Printhead Calculations

Pixel Pitch

pixel_{Pitch}=

sub_pixel_{Pitch}=

In [2]:

```
resolutions    = [360, 600, 1200, 2400, 4800] # dpi
inch2cm        = 2.54 # cm/inch
nbrOfSubpixels = 32

# Calulation Pixel Pinch
pixel_pitch = np.empty(shape=[size(resolutions)], dtype=np.float64) # um
for i in range(size(resolutions)):
  pixel_pitch[i] = (inch2cm/resolutions[i])*10000

# Calculation Subpixel Pinch
sub_pixel_pitch = np.empty(shape=[size(resolutions)], dtype=np.float64) # um
for i in range(size(resolutions)):
  sub_pixel_pitch[i] = pixel_pitch[i]/nbrOfSubpixels

for i in range(size(resolutions)):
  print("Resolution: {:4} dpi   Pixel Pitch: {} um   Sub Pixel Pitch: {} um".format(resolutions[i], pixel_pitch[
i], sub_pixel_pitch[i]))
```

```
Resolution:  360 dpi   Pixel Pitch: 70.5555555556 um   Sub Pixel Pitch: 2.20486111111 um
Resolution:  600 dpi   Pixel Pitch: 42.3333333333 um   Sub Pixel Pitch: 1.32291666667 um
Resolution: 1200 dpi   Pixel Pitch: 21.1666666667 um   Sub Pixel Pitch: 0.661458333333 um
Resolution: 2400 dpi   Pixel Pitch: 10.5833333333 um   Sub Pixel Pitch: 0.330729166667 um
Resolution: 4800 dpi   Pixel Pitch: 5.29166666667 um   Sub Pixel Pitch: 0.165364583333 um
```

Error in Pixel Pitch \Delta x_{Pitch}because of substrate speed and clock frequency x_{Pitch}=v_{Substrate}·t

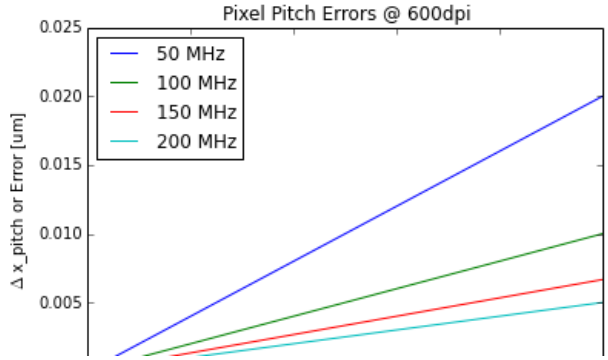
Error is the modulus of the achieved and the ideal pixel pitch. e=Rest()

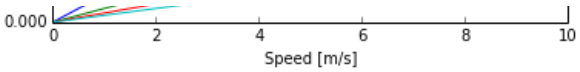
In [97]:

```
import numpy as np
import pylab as pl
pl.clf()
frequencies = [50e6, 100e6, 150e6, 200e6] # MHz
speed = np.linspace(0,10,50) # 50 x points from 0m/s to 10m/s

for freq in frequencies:
  delta_x_pitch = (speed*100000) * 1/freq # um/s / s
  #error = numpy.mod(pixel_pitch[1],delta_x_pitch)
  # Plot graph
  pl.plot(speed,delta_x_pitch,label=str(int(freq/1000000))+" MHz")

# Place legend, Axis and Title
pl.legend(loc='best')
pl.xlabel("Speed [m/s]")
pl.ylabel("$\Delta$ x_pitch or Error [um]")
pl.title("Pixel Pitch Errors @ {}dpi".format(resolutions[1]))
```

Out[97]:

```
<matplotlib.text.Text at 0x15df70f0>
```





Resolution

In [4]:

```
pixel_pitch = 0.1692 # mm
inch2cm = 2.54 # cm/inch
resolution = (inch2cm/pixel_pitch)*10
print("Resolution: {:4} dpi   Pixel Pitch: {} um".format(resolution, pixel_pitch))
```

```
Resolution: 150.11820331 dpi   Pixel Pitch: 0.1692 um
```

Stitching

For Steinemann machine (KonicaMinolta KM1024i)

pixel_{overlap}=

In [9]:

```
ph_resolution  = 360   # dpi
inch2cm        = 2.54  # cm/inch
ph_overlapping = 0.196 # cm

pixel_overlap = ph_resolution / inch2cm * ph_overlapping

print("{} dots".format(pixel_overlap))
```

```
27.7795275591 dots
```

Pixel distance

KonicaMinolta KM1024i & KM1024

pixel_{pitch}=

Printing Speed

pixel_{pitch} = = [mm]

freq_{adjusted} = = [Hz]

speed = pixel_{pitch} * freq_{adjusted} * = []

In [30]:

```
resolution_job = 720  # dpi
resolution_ph  = 360  # dpi
f_ph           = 30e3 # Hz
speed_percent  = 100  # %
#-------------------------------------
inch2mm     = 25.4 # mm/inch
#-------------------------------------
pixel_pinch = inch2mm/resolution_ph          # (inch/mm)/(dot/inch) = mm/dot
adjusted_frequency = f_ph / 100 * speed_percent # Hz

speed = pixel_pinch * adjusted_frequency * resolution_ph/resolution_job # (mm/dot) * 1/s * (dot/inch)/(dot/inch)
= mm/s

print("Pixel Pinch = {} mm".format(pixel_pinch))
print("Adjusted Frequency = {}% of {} Hz = {} Hz".format(speed_percent, f_ph, adjusted_frequency))

print("Speed = {} mm/s".format(speed))
print("Speed = {} m/s".format(speed/1000))
```

```
Pixel Pinch = 0.0705555555556 mm
Adjusted Frequency = 100% of 30000.0 Hz = 30000.0 Hz
Speed = 1058.33333333 mm/s
Speed = 1.05833333333 m/s
```

Printing Dimensions

For calculating the Printing dimensions in x and y axis. The following values are needed. * section_witdh = Number of bytes of a section. Can be found in CoE Memory Organization 8040:1 * section_depth = Number of bytes of a section. Can be found in CoE Memory Organization 8040:2 * Number of sections = Number of bytes of a section. Can be found in CoE Memory Organization 8040:3 * Resolution of the Printhead

print_{width} = section_{width} * 2\frac{pixel}{byte} * \frac{0.0254\frac{m}{inch}}{ph_{dpi}}

print_{length} = section_{depth} * section_{numbers} * 2\frac{pixel}{byte} * \frac{0.0254\frac{m}{inch}}{ph_{dpi}}

In [22]:

```
def printing_dimensions(section_width, section_depth, section_numbers, ph_dpi):
  """
  printing_dimensions calculates the printing width and length of a printhead with the available memory
  param section_width   : (bytes) Can be found in CoE Memory Organization 8040:1
  param section_depth   : (bytes) Can be found in CoE Memory Organization 8040:2
  param section_numbers : Can be found in CoE Memory Organization 8040:3
  """
  # Constants
  inch2m     = 0.0254 # m/inch
  pixel_byte = 2      # pixel/byte

  # Calc Pixel Pitch
  pixel_pitch = inch2m/ph_dpi # in m

  # Calc Print width = Y Axis = Crossprint Axis
  section_width_pixel = section_width * pixel_byte
  print_width = section_width_pixel*pixel_pitch

  # Calc Print length = X Axis = Print Axis
  section_depth_pixel = section_depth * pixel_byte
  print_length = section_depth_pixel*section_numbers*pixel_pitch

  # Calc memory size
  memory_size_byte  = section_width*section_depth*section_numbers
  memory_size_pixel = memory_size_byte * pixel_byte

  # Print
  print("Memory Size           : {} MBytes".format(memory_size_byte/1000/1000))
  print("Memory Size           : {} Pixel".format(memory_size_pixel))
  print("PixelPitch @ {:3}dpi   : {} um".format(ph_dpi,pixel_pitch*1000*1000))
  print("Print width           : {} mm".format(print_width*1000))
  print("Print length          : {} m".format(print_length))

  return print_width, print_length
```

Techma 4

In [23]:

```
section_width   = 3072  # bytes
section_depth   = 16384 # bytes
section_numbers = 21    # Nbr
ph_dpi          = 600   # dpi
print_width, print_length = printing_dimensions(section_width, section_depth, section_numbers, ph_dpi)

# For 4 GB of RAM
print_length = print_length *4
print("Print length (4GB)    : {} m".format(print_length))
```

```
Memory Size           : 1056 MBytes
Memory Size           : 2113929216 Pixel
PixelPitch @ 600dpi   : 42.3333333333 um
Print width           : 260.096 mm
Print length          : 29.130752 m
Print length (4GB)    : 116.523008 m
```