YPF Format(ver 1.7)									
Name						Position	Len(bytes)	sampleData	description	
Version						0	2	11 00(dec:17)		
Width						+2	2	22 01(dec:290)		
Height						+2	2	11 01(dec:273)		
Depth						+2	4	03 00 00 00 (dec:3)		
hasPalette						+4	4	01 00 00 00 (dec:1)	0==false, 1==true	
Palette						+4	512		if(hasPalette==1)	
DataPosition						+512	4	51 02 00 00 (dec:593)	RealDataPosition(00 00 02 51)	
ImageSet Count						+4	2	01 00 (dec:1)	Imageset Count //imageset is consist of somedatas, frames, actions, states,state transes)	
	dummyinfo	dummy1len				+2	4	00 00	dummy data len	
		dummy1position				4	4		dummy data potision	if(Dummy1Len!=0)
		dummy1data(IDX 따로)				dummy1position	Dummy1Len		dummy is unknown data	if(Dummy1Len!=0)
	frameInfo	framecount				+Dummy11Len	2	01 00 (dec:1)	Frame is real image	
		frame#0	top			+2	2	01 ff (dec:-240)	bottom-top = frameHeight	
			left			+2	2	6f ff(dec:-145)	right-left = frameWidth	
			bottom			+2	2	21 00 (dec:33)	bottom-top = frameHeight	
			right			+2	2	91 00 (dec:145)	right-left = frameWidth	

			flag			+2	4	05 00 00 00 bytes = 0..0..0..0000010bits
								(hasAlpha,hasBase,hasDepth)
									if(1bit==0) hasAlpha=true(flag & 00001000)
									if(2bit==1) hasBase=true(flag & 00000100)
									if(0bit==1) hasDepth=true(flag & 00000001)	

			alphalen			+4	4	78 7a 00 00(dec:31352)		if(hasAlpha)
			alphaoffset			+4	4	00 00 00 00 (dec:0)	alphaDataIndex = dataPosition + alphaOffset	if(hasAlpha)
			baseoffset			+4	4	78 7a 00 00 (dec:31352)		if(hasBase)
			baselen			+4	4	D2 61 00 00 (dec:25042)	ver 14 no has BaseLen Data.(BaseLen = frameHeight* frameWidth)	if(hasBase && !v14)
			depthtype			+4	1	1		if(hasDepth && !v16 && !v14)
			depthval2			+1	2	00 80 (dec:32768)		if(hasDepth && !v16 && !v14)
			depthnearestdist			+2	2	46 00 (dec:70)		if(hasDepth && !v16 && !v14)
			depthoffset			+2	4	4a dc 00 00(dec:56394)		if(hasDepth && !v16 && !v14)
			depthsize			+4	4	fb 22 00 00 (dec:8827)		if(hasDepth && !v16 && !v14)
			dummy2len			4	4	05 00 00 00 (dec:5)	dummy2 data len	
			dummy2position			4	4	15 1f 00 00 (dec:8221)	dummy2 data position	if(Dummy2Len!=0)
			dummy2data(IDX 따로)			Dummy2position	Dummy2Len			if(Dummy2Len!=0)

			alphadata(Base까지 IDX 따로)			DataPosition+alphaOffset(593+0)	FrameHeight*4 ((33+240=273)*4=1092	alphaData[0]:00 00 00 00(dec:0) => rows[0]:0
								alphaData[1]:0c 00 00 00(dec:12) => rows[1]:6
								alphaData[2]:1c 00 00 00(dec:12) => rows[2]:8
								..
								alphaData[271]:20 76 00 00(dec:30240) => rows[271]:6
								alphaData[272]:2c 76 00 00(dec:30252) => rows[272]:6
									1)Get rows data
									rows[0] = alphaData[0]-0/2;
									rows[1] = alphaData[1]-alphaData[0]/2;
									..
									rows[frameHeight] = alphaData[frameHeight]-alphaData[frameHeight-1]/2;
										if(hasAlpha)

				alphamask[frameheight][framewidth] => default val:0x00 read until dataoffset + alphaoffset+alphalen	alphamask[0][0~115]:0 alphamask[0][116~framewidth-1]:0	+frameHeight*4	(11bit) data1(8bit) + data2(8bit)	data1:00000000 data2:01110011 bits => alphaMask[0].len = 000 01110011 bits = 115	alphaMask[0][0] = (data1&11111000) => data1's up 5 bits = 00000bits = 0 alphaMask[0][1] = data1&11111000) => data1's up 5 bits = 00000bits = 0 .. alphaMask[0][114] = data1&11111000) => data1's up 5 bits = 00000bits = 0	xxxxx yyy yyyyyyyy => x mean alphaMaskData , y mean alphaMaskLen
					alphamask[1][0]:8 alphamask[1][1~framewidth-1]:0	+2	(11bit) data1(8bit) + data2(8bit)	data1:00001000 data2:00000001 bits => alphaMask[1].len = 000 00000001 bits = 1	alphaMask[1][0] = (data1&11111000) => data1's up 5 bits = 01000bits = 8	
					..					
					alphamask[frameHeight]	+2	(11bit) data1(8bit) + data2(8bit)	data1:00000111 data2:11111111 bits	alphaMask[0+frameWidth*frameHeight]=data1&0xf8 alphaMask[1+frameWidth*frameHeight]=data1&0xf8 .. alphaMask[lineN.len+frameWidth*frameHeight]=data1&0xf8	

			Base(image coloring_		if(alphaMask[i]==0x00)
					{
					color[i*2] = 0x1f;
					color[i*2+1] = 0x8;
					}
					else if(baseOffset==0)
					{
					color[i*2] = 0x00;
					color[i*2+1] = 0x00;
					}
					else
					{if(hasPalete)
					{
					color[i*2]=pallete[imageBase[idx]*2];
					color[i*2+1]=pallete[imageBase[idx]*2+1];
					}
					else
					{
					color[i*2]=imageBase[idx++];
					color[i*2+1=imageBase[idx++];
					}
						DataPosition+BaseOffset(593+31352)	Baselen			Color Pallet's Index

		frame#1								
		..								
		frame#N								
	actionInfo(ani)	Action count				+2	2			
		action#0	action size			+2	4			
			action offset			+4	4			if(actionSize!=0)
			time1			+4	6			
			time2			+6	6			
			actionelementcount			+6	2			
			actionElement#0	frameindex		+2	2			
				time		+2	4			
				actionElementLen		+4	4			
				actionElementOffset		+4	4			if(actionElementLen!=0)
				offsetX(IDX 따로 X~dummy까지)		actionElementOffset	1			if(actionElementLen!=0)
				offsetY		+1	1			if(actionElementLen!=0)
				dummy		+1	2			if(actionElementLen!=0)
			..							
			actionElement#N							
		action#1								
		..								
		action#N								
	StateInfo	state count				+2	2			
		state#0	statesize			+2	4			
			stateoffset			+4	4			if(StateSize!=0)
			stateElementCount			+4	2			
			stateElemet#0	isStateElementFrame		+2	1			
				dummy1		+1	1			
				dummy2		+1	1			
				dummy3		+1	2			
				dummy4		+2	4			if(isStateElementFrame!=0)
				stateElementSize		+4	4			
				stateElementOffset		+4	4			if(StateElementSize!=)
		..								
		state#N								
	stateTrtansValue	StateTransValueCount				+4	2			
		stateTrans#0	key1			+2	2			
			key2			+2	2			
			value			+2	2			
			stateTranseValueSize			+2	4			
			stateTranseValueOffset			+4	4			if(StateTransValueSize!=0)
		stateTrans#1	key1			+4+StateTransValueOffset				
			key2							
			value							
			stateTranseValueSize							
			stateTranseValueOffset							if(StateTransValueSize!=0)
		..								
		stateTrans#N								
ImageSet#N										