

**ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

ĐINH CAO SƠN

**NGHIÊN CỨU, THIẾT KẾ NỀN TẢNG NHÚNG
THỰC THI CÁC ỨNG DỤNG XỬ LÝ ẢNH THỜI GIAN THỰC**

**LUẬN VĂN THẠC SĨ
NGÀNH CÔNG NGHỆ KỸ THUẬT ĐIỆN TỬ, TRUYỀN THÔNG**

HÀ NỘI – 2017

**ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

ĐINH CAO SƠN

**NGHIÊN CỨU, THIẾT KẾ NỀN TẢNG NHÚNG
THỰC THI CÁC ỨNG DỤNG XỬ LÝ ẢNH THỜI GIAN THỰC**

Ngành : Công nghệ kỹ thuật điện tử, truyền thông

Chuyên ngành : Kỹ thuật điện tử

Mã số : 60520203

**LUẬN VĂN THẠC SĨ
NGÀNH CÔNG NGHỆ KỸ THUẬT ĐIỆN TỬ, TRUYỀN THÔNG**

NGƯỜI HƯỚNG DẪN KHOA HỌC: TS. PHẠM MINH TRIỂN

HÀ NỘI - 2017

LỜI CẢM ƠN

Thời gian 2 năm học đối với chương trình đào tạo thạc sĩ có thể là dài đối với một số học viên, nhưng đối với em – một kỹ sư vừa đi học vừa đi làm thì quãng thời gian đó là khá ngắn ngủi. Cộng với đó là lịch làm việc dày đặc tại Viettel R&D nên em cũng đã thực sự nỗ lực và cố gắng để hoàn thành luận văn này. Tuy nhiên, quãng thời gian quý báu với những giờ lên lớp vào các buổi tối và các ngày cuối tuần đó đã giúp trang bị cho em rất nhiều tri thức bổ ích, giúp em củng cố các khái niệm, các nguyên lý của điện tử tương tự, điện tử số và điều khiển. Những bài giảng của thầy Trần Quang Vinh, thầy Bạch Gia Dương, thầy Trần Đức Tân, thầy Phạm Minh Triễn và nhiều thầy cô khác đã in sâu trong trí nhớ và là những chiếc chìa khóa mở đầu cho con đường nghiên cứu chuyên nghiệp của Em hiện tại và trong tương lai. Em xin gửi lời cảm ơn chân thành đến các thầy, kính chúc các thầy sức khỏe dồi dào và luôn cháy bỏng đam mê, cống hiến cho khoa học, giúp truyền đạt những tri thức quý báu cho lớp lớp học viên như chúng em.

Ngoài những cố gắng của bản thân để hoàn thiện luận văn này, em còn nhận được sự giúp đỡ tận tình từ thầy giáo hướng dẫn – **TS. Phạm Minh Triễn**, người luôn đặt ra các yêu cầu cao, các bài toán khó và cùng em bàn bạc giải quyết các vấn đề không chỉ trong phạm vi luận văn này mà còn là các vấn đề về định hướng, hỗ trợ sinh viên nghiên cứu khoa học, các vấn đề hợp tác giữa Trường Đại học và Công ty. Giúp tìm đầu ra cho các sinh viên sau khi tốt nghiệp. Một người thầy với tầm nhìn và hoài bão lớn, rất đáng để em học tập, noi gương.

Luận văn này được tôi thực hiện song song với một đề tài nghiên cứu tại Viện Nghiên cứu và Phát triển Viettel do tôi làm chủ nhiệm. Tại đây, tôi cùng các cộng sự đã cố gắng để tạo nên một sản phẩm Quân sự “Made in Vietnam”. Kết quả nghiên cứu của đề tài này, một phần cũng đã được áp dụng vào sản phẩm đó. Tôi cũng xin gửi lời cảm ơn chân thành tới ban lãnh đạo đơn vị và các đồng nghiệp thân yêu đã giúp đỡ, cùng tôi giải quyết những vấn đề khó để đạt được những kết quả bước đầu đáng khích lệ.

Qua đây, Em xin gửi lời cảm ơn chân thành tới BCN Khoa Điện tử Viễn thông, BGH và các phòng ban chức năng của Đại học Công nghệ - ĐHQG Hà Nội, đã tạo điều kiện giúp đỡ trong suốt quãng thời gian em theo học chương trình đào tạo thạc sĩ tại trường.

Cuối cùng, xin gửi lời cảm ơn chân thành tới bố, mẹ, vợ và các con thân yêu đã luôn tin tưởng, ủng hộ và động viên trong quá trình theo học chương trình đào tạo thạc sĩ cũng như trong thời gian tập trung thực hiện đề tài này.

Thân ái,

Đinh Cao Sơn

LỜI CAM ĐOAN

Tôi xin cam đoan đề tài “Nghiên cứu, thiết kế nền tảng nhúng thực thi các ứng dụng xử lý ảnh” do tôi trực tiếp thực hiện. Không sao chép lại từ bất kỳ một nguồn nào (*sách, báo, tạp chí, công trình khoa học, ...*) trong và ngoài nước đã công bố.

Nếu phát hiện thấy bất kỳ một vi phạm nào về bản quyền các nội dung tôi đề cập trong đề tài, tôi xin chịu hoàn toàn trách nhiệm trước hội đồng đánh giá và nhà trường.

Hà Nội, ngày 11 tháng 11 năm 2017

Đinh Cao Sơn

MỤC LỤC

DANH MỤC CÁC CHỮ VIẾT TẮT	4
DANH MỤC CÁC BẢNG BIỂU.....	5
DANH MỤC CÁC HÌNH VẼ	6
MỞ ĐẦU.....	7
CHƯƠNG 1: TỔNG QUAN VỀ THIẾT BỊ XỬ LÝ ẢNH THỜI GIAN THỰC	8
1. Thiết bị xử lý ảnh bắt bám mục tiêu Video Tracker 8200 – EOImaging (Mỹ).....	8
2. KIT phát triển xử lý ảnh và streaming video Z3-DM8169-VI-RPS (Mỹ).....	9
3. Đề xuất thiết kế thiết bị xử lý ảnh thời gian thực	9
CHƯƠNG 2: YÊU CẦU VÀ THIẾT KẾ CHI TIẾT PHẦN CỨNG HỆ THỐNG....	12
1. Khối đầu vào video	13
1.1. Đầu vào tương tự.....	13
1.2. Đầu vào số	17
2. Khối xử lý	18
3. Bộ nhớ	25
3.1. Bộ nhớ lưu trữ	25
3.2. Bộ nhớ đệm	25
4. Đầu ra video.....	26
5. Khối cấp nguồn.....	27
6. Gia công chế tạo.....	29
CHƯƠNG 3: ĐÁNH GIÁ HỆ THỐNG VỚI THUẬT TOÁN MẪU.....	31
1. Giới thiệu thuật toán.....	31
1.1. Đặt vấn đề.....	31
1.2. Mô tả thuật toán.....	32
1.3. Nội dung và lưu đồ xử lý thuật toán.....	33
2. Triển khai thuật toán	37
3. Một số hình ảnh thử nghiệm.....	38
KẾT LUẬN	39
TÀI LIỆU THAM KHẢO.....	40
PHỤ LỤC: HỒ SƠ THIẾT KẾ NGUYÊN LÝ SẢN PHẨM.....	42

DANH MỤC CÁC CHỮ VIẾT TẮT

Thuật ngữ/Viết tắt	Ý nghĩa
ASIC	Application-Specific Integrated Circuit
DSP	Digital Signal Processing
FPGA	Field Programmable Gate Array
HDMI	High Definition Media Interface
NTSC	National Television System Committee
PAL	Phase Alternating Line
RGA	Running Gaussian Average
SDI	Serial Digital Interface

DANH MỤC CÁC BẢNG BIỂU

Bảng 1. 1: Tốc độ bitrate tương ứng với các chuẩn video	11
Bảng 2. 1: Yêu cầu thiết kế chi tiết phần cứng hệ thống.....	12
Bảng 2. 2: Mối liên hệ giữa độ phân giải ảnh, tần số quét màn hình và tần số pixel tối đa	15
Bảng 2. 3: So sánh giữa các dòng chip của hãng Texas Instrument.	19
Bảng 2. 4: Danh sách nguồn cấp cho chip DSP	28
Bảng 3. 1: Kết quả đánh giá thuật toán phát hiện chuyển động.....	37

DANH MỤC CÁC HÌNH VẼ

Hình 1. 1: Thiết bị xử lý ảnh Video Tracker 8200	8
Hình 1. 2: KIT phát triển xử lý ảnh và streaming video của Z3 Technology	9
Hình 1. 3: Kiến trúc lõi xử lý chiếc máy tính nhúng đầu tiên mang tên AGC	10
Hình 1. 4: Sơ đồ khối kiến trúc xử lý của Apollo17	10
Hình 2. 1: Các chuẩn video tương tự.....	13
Hình 2. 2: Dạng sóng của tín hiệu video tương tự	14
Hình 2. 3: Minh họa biến đổi tín hiệu video gốc sang composite video	14
Hình 2. 4: Minh họa bức ảnh đen trắng kích thước $N_H \times N_V$	15
Hình 2. 5: Nguyên lý khối đọc đầu vào tương tự composite.....	16
Hình 2. 6: Các chuẩn video số.....	17
Hình 2. 7: Nguyên lý khối đọc đầu vào số HDMI	18
Hình 2. 8: Biểu đồ đánh giá thể mạnh của các nền tảng	19
Hình 2. 9: Kiến trúc chip xử lý TMS320DM8168	21
Hình 2. 10: Bố trí chân của chip TMS320DM8168	22
Hình 2. 11: Kiến trúc lõi xử lý ARM	22
Hình 2. 12: Kiến trúc lõi xử lý DSP	23
Hình 2. 13: Kiến trúc CPU C674x.....	24
Hình 2. 14: Nguyên lý kết nối đến bộ nhớ SPI Flash.....	25
Hình 2. 15: Mô hình kết nối DDR3 8 bit.....	26
Hình 2. 16: Nguyên lý kết nối IC giao tiếp Ethernet.....	27
Hình 2. 17: Thứ tự khởi động của các nguồn cấp cho DSP	28
Hình 2. 18: Bản vẽ gia công bo mạch thử nghiệm	29
Hình 2. 19: Bản vẽ lắp ráp linh kiện mặt trên	30
Hình 2. 20: Bản vẽ lắp ráp linh kiện mặt dưới	30
Hình 3. 1: Minh họa phát hiện chuyển động	31
Hình 3. 2: Mô hình của các thuật toán Background subtraction	32
Hình 3. 3: Sơ đồ thuật toán phát hiện chuyển động	33
Hình 3. 4: Phát hiện trong điều kiện thiếu sáng	38
Hình 3. 5: Phát hiện với ảnh nhiệt.....	38
Hình 4. 1: Minh họa máy tính nhúng dùng trong Quân sự	39
Hình 4. 2: Minh họa ứng dụng cho máy bay UAV	39

MỞ ĐẦU

1. THỰC TRẠNG CÔNG NGHỆ TẠI THỜI ĐIỂM NGHIÊN CỨU

a. Việt Nam

Công nghệ xử lý ảnh (Image Processing) hay sau này gọi là thị giác máy tính (Computer Vision) đã được nghiên cứu tại Việt Nam từ khá sớm. Từ việc triển khai các giải thuật trên Matlab cho đến các ứng dụng thực tế. Tuy nhiên, đa phần nhiệm vụ này được thực thi trên một máy tính (PC hoặc laptop) sử dụng các thư viện kinh điển như OpenCV.

Việc triển khai các thuật toán xử lý ảnh trên nền tảng một vi xử lý tính toán chuyên dụng còn khá mới mẻ và ít người tiếp cận.

b. Thế giới

Cùng với sự phát triển của công nghệ bán dẫn, mật độ tích hợp ngày càng cao. Các hãng sản xuất chip lớn đã cho ra đời các vi xử lý có khả năng tính toán dấu phẩy động, tốc độ tính toán lên đến hàng nghìn triệu lệnh trong một giây (> 1000 MIPS) từ những năm 2010.

Kể từ đó, người ta đã quan tâm nhiều hơn đến việc đưa các thuật toán tính toán phức tạp lên các vi xử lý nhỏ gọn, tiêu tốn ít năng lượng để tạo ra các nền tảng xử lý thông minh, hỗ trợ con người nhiều hơn trong các ứng dụng xử lý hình ảnh.

2. NHIỆM VỤ - NỘI DUNG NGHIÊN CỨU ĐỀ TÀI ĐỀ XUẤT

Đề tài này được thực hiện với mục đích thiết kế, chế tạo thành công một nền tảng xử lý tính toán mạnh mẽ, hỗ trợ thực thi các thuật toán xử lý ảnh phức tạp, tiêu tốn công suất thấp, nhỏ gọn thay cho nền tảng máy tính thông thường.

Từ những kết quả tìm hiểu, phân tích sản phẩm của các hãng lớn trên thế giới. Tác giả đề xuất nội dung nghiên cứu chính của đề tài là nghiên cứu, thiết kế nền tảng phần cứng nhúng cho phép thực thi các ứng dụng xử lý ảnh, đáp ứng thời gian thực. Nội dung đề tài được chia thành 3 chương:

CHƯƠNG 1: TỔNG QUAN VỀ THIẾT BỊ XỬ LÝ ẢNH THỜI GIAN THỰC

CHƯƠNG 2: YÊU CẦU VÀ THIẾT KẾ CHI TIẾT PHẦN CỨNG HỆ THỐNG

CHƯƠNG 3: ĐÁNH GIÁ HỆ THỐNG VỚI THUẬT TOÁN MẪU

Những nội dung trên được cụ thể hóa trong luận văn này như sau:

CHƯƠNG 1: TỔNG QUAN VỀ THIẾT BỊ XỬ LÝ ẢNH THỜI GIAN THỰC

Thiết bị xử lý ảnh thời gian thực được đề xuất nghiên cứu, chế tạo phục vụ việc xây dựng nền tảng phần cứng nhúng có tài nguyên và năng lực xử lý đủ mạnh để thực hiện một loạt các bài toán ứng dụng trong lĩnh vực thị giác máy tính như tiền xử lý ảnh, lọc nhiễu, phân tích video hay chống rung hình ảnh.

Trong quá trình định nghĩa sản phẩm, tác giả đã tham khảo rất nhiều các cấu hình sản phẩm khác nhau của các hãng trên thế giới. Trong đó, nổi bật là thiết bị bắt bám mục tiêu tốc độ cao Video Tracker 8200⁽¹⁾ của EOImaging (Mỹ), thiết bị xử lý ảnh trên nền tảng DSP IGEPv2⁽²⁾ của ISEE (Tây Ban Nha), KIT phát triển xử lý ảnh OZ745⁽³⁾ của Omnitek (Anh), KIT phát triển Z3-DM8169-VI-RPS⁽⁴⁾ của Z3 Technology (Mỹ).

Trong số đó, ta có thể phân tích hai sản phẩm tiêu biểu là:

1. Thiết bị xử lý ảnh bắt bám mục tiêu Video Tracker 8200 – EOImaging (Mỹ)

Thiết bị được thiết kế chuyên dụng cho mục đích bắt bám mục tiêu. Có khả năng bắt bám theo nhiều đặc trưng khác nhau như mức xám (intensity), đường bao (edge), so khớp tương đồng (correlation), vv... Do đó, phù hợp với nhiều bài toán bắt bám khác nhau, trong từng điều kiện môi trường, với từng đối tượng cụ thể.



Hình 1. 1: Thiết bị xử lý ảnh Video Tracker 8200

Một đặc điểm vượt trội của sản phẩm này là có băng thông đầu vào lớn, có thể xử lý với tối đa bitrate đầu vào lên đến 3Gbps (tương ứng với độ phân giải 1080p, tốc độ 60 hình/giây), đồng thời có thể xử lý với cả đầu vào video dạng tương tự (PAL/NTSC) hoặc dạng số 3G-SDI.

Các thành phần xử lý chính của sản phẩm là chip DSP mã TMS320C6455 của Texas Instruments. Đồng thời để xử lý dữ liệu đầu vào tốc độ cao, một module FPGA (Spartan 6 XC6SLX100) được sử dụng để thực hiện đọc và tiền xử lý dữ liệu trước khi đưa vào xử lý tinh trong DSP. Module FPGA này cũng làm nhiệm vụ điều khiển

việc xuất dữ liệu ra những chuẩn tương tự, số, tương ứng với đầu vào. Trong khi xử lý với tốc độ rất cao thì kích thước của thiết bị chỉ bằng 114x127 (mm), và công suất tiêu thụ chỉ vào khoảng 12W.

2. KIT phát triển xử lý ảnh và streaming video Z3-DM8169-VI-RPS (Mỹ)



Hình 1. 2: KIT phát triển xử lý ảnh và streaming video của Z3 Technology

KIT phát triển Z3-DM8169-VI-RPS có kiến trúc gồm 2 chip xử lý chính là DSP DM8168 và FPGA Spartan 6.

Sản phẩm hỗ trợ các đầu vào video: Component, HDMI, 3G-SDI, đầu ra HDMI, 3G-SDI.

Khối xử lý gồm 1 lõi ARM cortex A8 tốc độ 1.2 GHz và 1 lõi DSP C674x tốc độ 1 GHz.

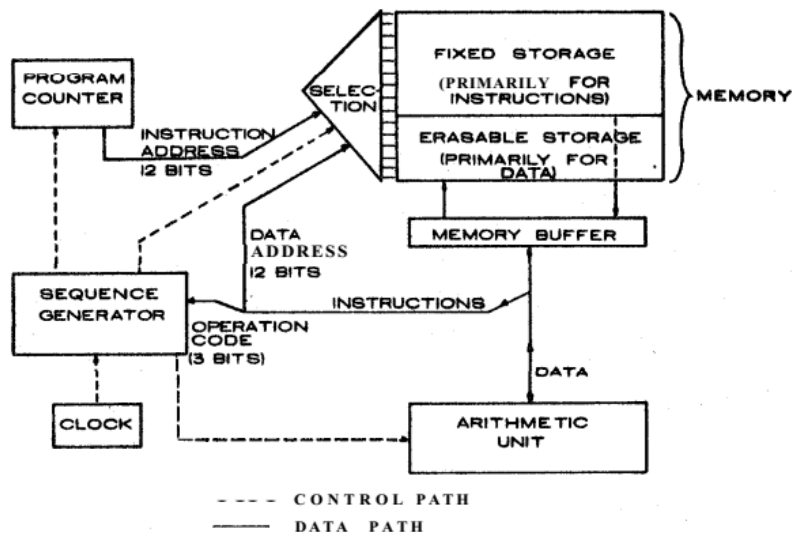
Về tính năng, ngoài các tính năng xử lý ảnh mà thư viện TI hỗ trợ, KIT còn cho phép các tính năng mã hóa H.264 và MPEG-2. Xử lý với độ phân giải 1080p/i 60fps.

Kích thước bo mạch là 218x90 (mm).

Công suất hoạt động: 14W.

3. Đề xuất thiết kế thiết bị xử lý ảnh thời gian thực

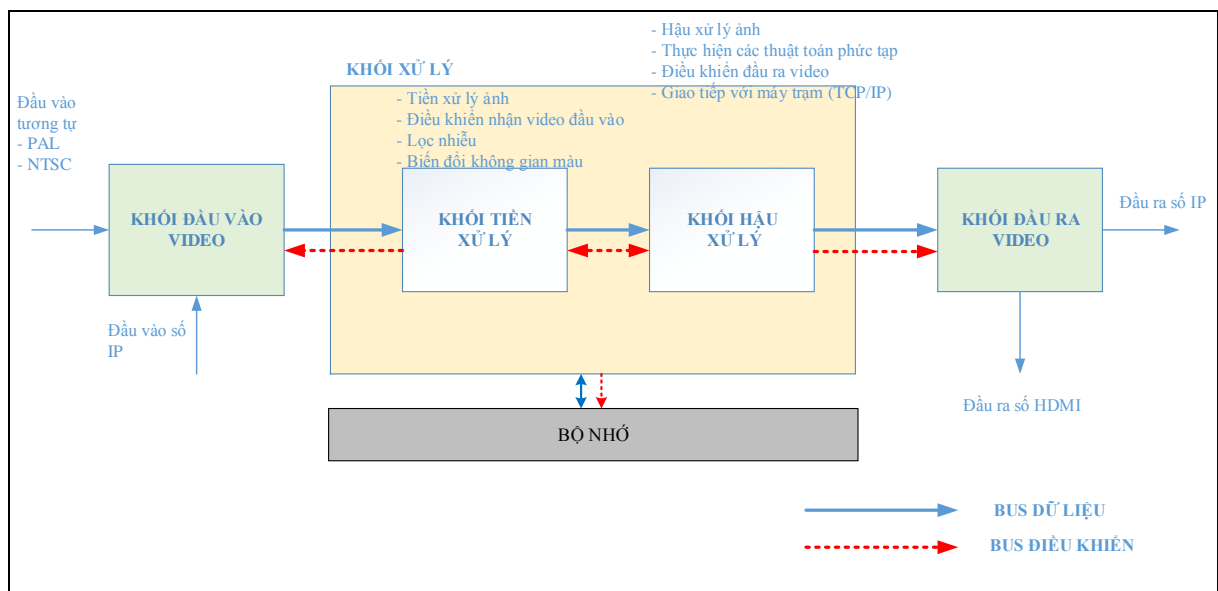
Qua phân tích các sản phẩm tham khảo tiêu biểu, tác giả đã định nghĩa sản phẩm thiết bị xử lý ảnh trên nền tảng nhúng với tên gọi **Apollo17**, được lấy theo tên gọi chương trình đưa con người lên mặt trăng của NASA mang tên chương trình Apollo khởi xướng vào những năm 1960. Trong đó, NASA đã phát triển một chiếc máy tính dẫn đường mang tên Apollo Guidance Computer (AGC). Đóng vai trò là bộ não của hệ thống.



Hình 1.3: Kiến trúc lõi xử lý chiếc máy tính nhúng đầu tiên mang tên AGC

Sơ đồ khối kiến trúc xử lý của Apollo17 được đề xuất thiết kế gồm 4 khối chính:

- Khối đầu vào video
- Khối xử lý: Gồm tiền xử lý và hậu xử lý
- Bộ nhớ
- Khối đầu ra video



Hình 1.4: Sơ đồ khối kiến trúc xử lý của Apollo17

Trong đó,

a. Đầu vào video

Nhận luồng video đầu vào từ đầu vào tương tự chuẩn PAL/NTSC hoặc từ đầu vào số IP.

b. Khối xử lý

Khối xử lý thực thi nhiệm vụ xử lý tín hiệu hình ảnh thu nhận được từ đầu vào. Đảm bảo việc xử lý đáp ứng thời gian thực (thời gian xử lý phải nhỏ hơn 40ms, tương đương 25 hình/giây).

Khối xử lý được chia thành 2 loại: Tiền xử lý và hậu xử lý.

- Tiền xử lý: Thực hiện điều khiển việc đọc dữ liệu video đầu vào từ các ADC hoặc thông qua giao thức Ethernet. Thực hiện các thao tác tiền xử lý ảnh (lọc nhiễu, biến đổi không gian màu, ...)
- Hậu xử lý: Thực hiện các thuật toán phức tạp; điều khiển đầu ra video và giao tiếp với máy trạm qua TCP/IP.

c. Bộ nhớ

Bộ nhớ sử dụng gồm 2 loại: bộ nhớ dùng để lưu trữ hệ điều hành và các tham số cấu hình hệ thống; bộ nhớ đệm dữ liệu phục vụ thực thi hệ điều hành và các tác vụ xử lý ảnh và điều khiển.

d. Đầu ra video

Tín hiệu video sau khi qua khối xử lý ảnh được nén và truyền ra ngoài qua IP hoặc hiển thị ra màn hình qua kết nối HDMI.

Bảng sau cho biết tốc độ dữ liệu trao đổi xuyên suốt hệ thống tương ứng với các độ phân giải ảnh (chuẩn) khác nhau.

Bảng 1. 1: Tốc độ (bitrate) tương ứng với các chuẩn video

Chuẩn	Kích thước (pixel)		Tốc độ frame (fps)	Tốc độ bitrate (bps)	Tương đương
	W	H			
VGA	640	480	25	184,320,000	175.78Mbps
			30	221,184,000	210.94Mbps
			60	442,368,000	421.88Mbps
			100	737,280,000	703.13Mbps
D1	720	576	25	248,832,000	237.3Mbps
			30	298,598,400	284.77Mbps
			60	597,196,800	569.53Mbps
			100	995,328,000	949.22Mbps
HD 720p	1280	720	25	552,960,000	0.51Gbps
			30	663,552,000	0.62Gbps
			60	1,327,104,000	1.24Gbps
			100	2,211,840,000	2.06Gbps
HD 1080p	1920	1080	25	1,244,160,000	1.16Gbps
			30	1,492,992,000	1.39Gbps
			60	2,985,984,000	2.78Gbps
			100	4,976,640,000	4.63Gbps

CHƯƠNG 2: YÊU CẦU VÀ THIẾT KẾ CHI TIẾT PHẦN CỨNG HỆ THỐNG

Yêu cầu chi tiết đối với từng khối chức năng của Apollo17, cụ thể như trong bảng 2.1:

Bảng 2. 1: Yêu cầu thiết kế chi tiết phần cứng hệ thống

TT	Tên khối	Yêu cầu		Linh kiện tham khảo (Mã NSX/Hãng/Xuất xứ)
		SL	Thông số kỹ thuật	
a	Đầu vào video			
1	Đầu vào video SD	1	CVBS	TVP5150AM/TI/Mỹ
2	Đầu vào video HD (tương tự)	1	YPrPb	TVP7002/TI/Mỹ
3	Đầu vào video HD (số)	1	IP	KSZ9031MNXIA/Microchip/ Mỹ
b	Khối xử lý			
4	Chip xử lý	1	<ul style="list-style-type: none"> - DSP clock: > 600MHz - Hỗ trợ tính toán dấu phẩy động - Tốc độ tính toán: > 4000 MIPS. - ARM clock: > 800MHz - Đầu vào video: hỗ trợ 2 đầu vào song song - Đầu ra video: hỗ trợ các đầu ra tương tự, số với các độ phân giải khác nhau. 	TMS320DM8168CCYG2/TI/ Mỹ
c	Bộ nhớ			
5	SPI Flash	1	- 64Mbit	W25Q64FVSSIG /Winbond/Đài Loan
6	DDR3	4	- 256Mbyte	MT41J256M8DA-093:K /Micron/Mỹ
7	NAND Flash	1	- 512Mbyte (4Gbit)	MT29F4G16ABADAH4- IT:D/Micron/Mỹ
8	Bộ nhớ eMMC	1	- \geq 4GB	MTFC4GACAANA-4M IT/Micron/Mỹ
d	Đầu ra video			
9	Đầu ra số IP	1	IP	KSZ9031MNXIA/Microchip/ Mỹ
10	Đầu ra hiển thị HDMI	1	HDMI 1.3	TPD12S521DBTR/TI/Mỹ
e	Kênh kết nối ngoại vi			
11	Kết nối Ethernet	2	- 1Gbps, GMII	KSZ9031MNXIA/Microchip/ Mỹ

12	RS232/RS422	3	2xRS232C 1xRS422	MAX3232CPWR/MAX3488E ESA+T/TI/Mỹ
13	USB host OTG	1	Giao diện joystick/keyboard/mouse	
14	Led báo nguồn	1	LED 1 màu RED	
15	LED báo trạng thái	2	LED 1 màu BLUE	
16	Đầu ra hiển thị HDMI	1	HDMI 1.3	TPD12S521DBTR/TI/Mỹ
f	Khối nguồn	1		
17	- Dải điện áp: 5.5-36V/3A. - Pwr tree: 5V, 3.3V, 2.5V, 1.8V, 1.5V, 1.2V, 0.75V - Sequence: Continuously - Protection: short-circuit, fuse 3A - Công suất trung bình < 15W	1	Đảm bảo cấp nguồn cho bo mạch, các linh kiện đặc thù (chip, RAM) và ngoại vi kết nối vào.	TPS5450QDDARQ1/TPS65253RHDR/TPS22918DBVR/TPS22810DBVR/TPS54620RGYR/TPS77018DBVR/TPS51200DRCR/TPS3808G09DBVRG4/TI/Mỹ

1. Khối đầu vào video

Chúng ta biết rằng, tín hiệu video được chia thành 2 loại chính phân theo mục đích: video quảng bá, phục vụ truyền hình, giải trí (broadcast) và video đồ họa, phục vụ các ứng dụng hình ảnh trên PC, thiết bị chuyên dụng (graphics).

Có rất nhiều các định dạng video tương tự và số khác nhau cùng tồn tại song song hiện nay.

Việc thiết kế khối video input phải tuân theo một số nguyên tắc nhất định như phân tích dưới đây.

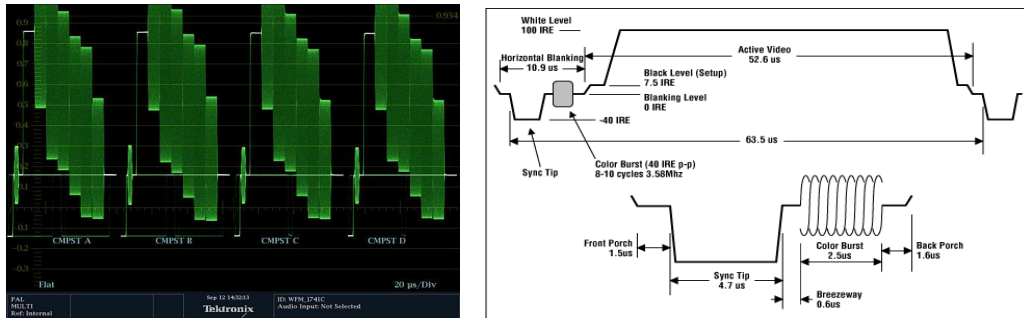
1.1. Đầu vào tương tự

- Tín hiệu video tương tự được chia thành 3 chuẩn giao tiếp cơ bản: Composite hay CVBS (1 dây); S-Video (2 dây) và Component (3 dây).



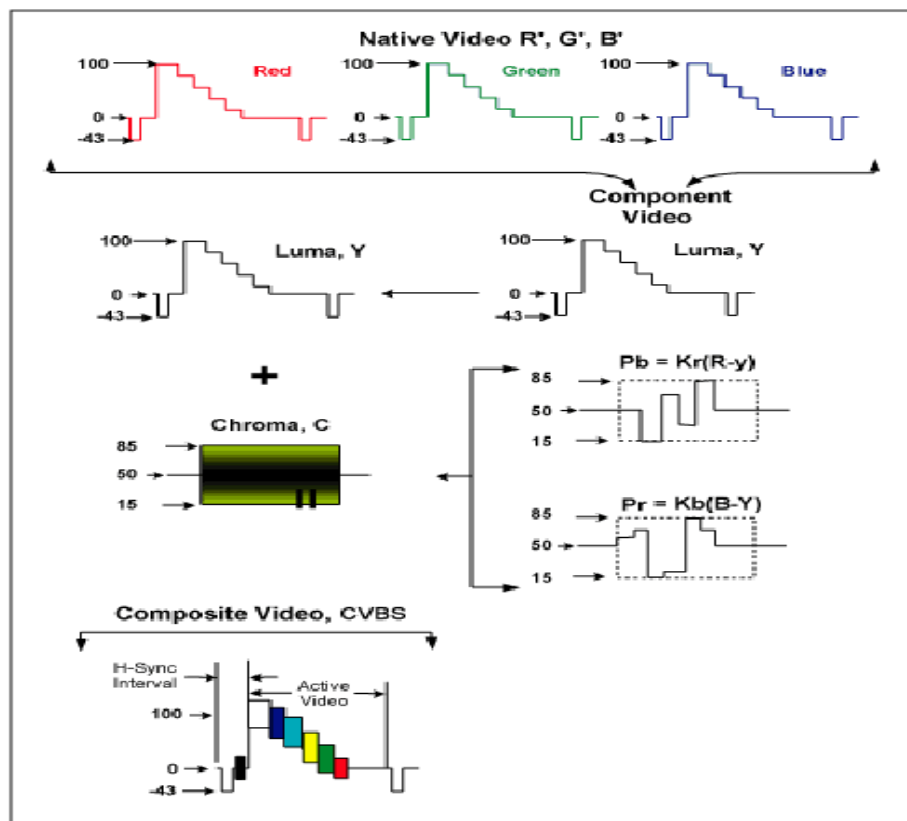
Hình 2. 1: Các chuẩn video tương tự

- Dạng sóng của các tín hiệu video tương tự như sau:



Hình 2. 2: Dạng sóng của tín hiệu video tương tự

Việc biến đổi để tín hiệu tương tự để cho ra các định dạng khác nhau được minh họa như hình sau:



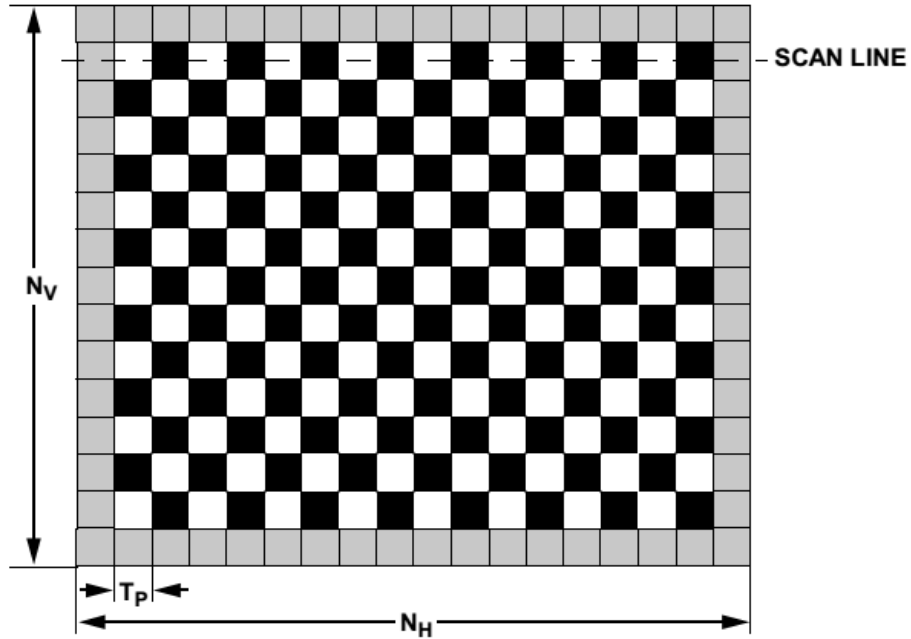
Hình 2. 3: Minh họa biến đổi tín hiệu video gốc sang composite video

- Việc thu nhận tín hiệu video tương tự được thực hiện bằng cách sử dụng các ADC chuyên dụng (như TVP5150AM của Texas Intrusment; MAX9526 của Maxim Integrated hay ADV7403 của Analog Device).
- Các ADC này sẽ thực hiện quá trình lấy mẫu, lượng tử và mã hóa tín hiệu để cho ra đầu ra số của tín hiệu tương tự.
- Ngoài ra, đối với tín hiệu video nói chung, cần chú ý đến việc hiển thị ra màn hình. Và một trong những tham số quan trọng là tần số quét tín hiệu (fv). Tần số fv càng lớn, bức ảnh hiển thị ra càng mịn.

Ví dụ: Ta xét một bức ảnh có kích thước $N_H \times N_V$ như hình sau:

Trong đó:

- N_H là tổng số điểm ảnh (pixel) theo chiều ngang
- N_V là tổng số điểm ảnh theo chiều dọc
- T_p là thời gian quét qua 1 điểm ảnh
- T_v là thời gian quét tổng của bức ảnh



Hình 2. 4: Minh họa bức ảnh đen trắng kích thước $N_H \times N_V$

Khi đó, thời gian quét cho một pixel được tính như sau:

$$T_p = \frac{T_v}{N_H \times N_V} \quad (2.1)$$

Do đó, nếu coi một chu kỳ đầy đủ mất 2 lần thời gian T_p (quét qua 2 pixel), thì tần số quét cho một pixel tối đa (tần số pixel) sẽ là:

$$f_{max} = \frac{1}{2 \times T_p} = \frac{N_H \times N_V}{2 \times T_v} = \frac{N_H \times N_V \times f_v}{2} \quad (2.2)$$

Với $f_v = 1/T_v$, là tần số quét màn hình (fresh-rate).

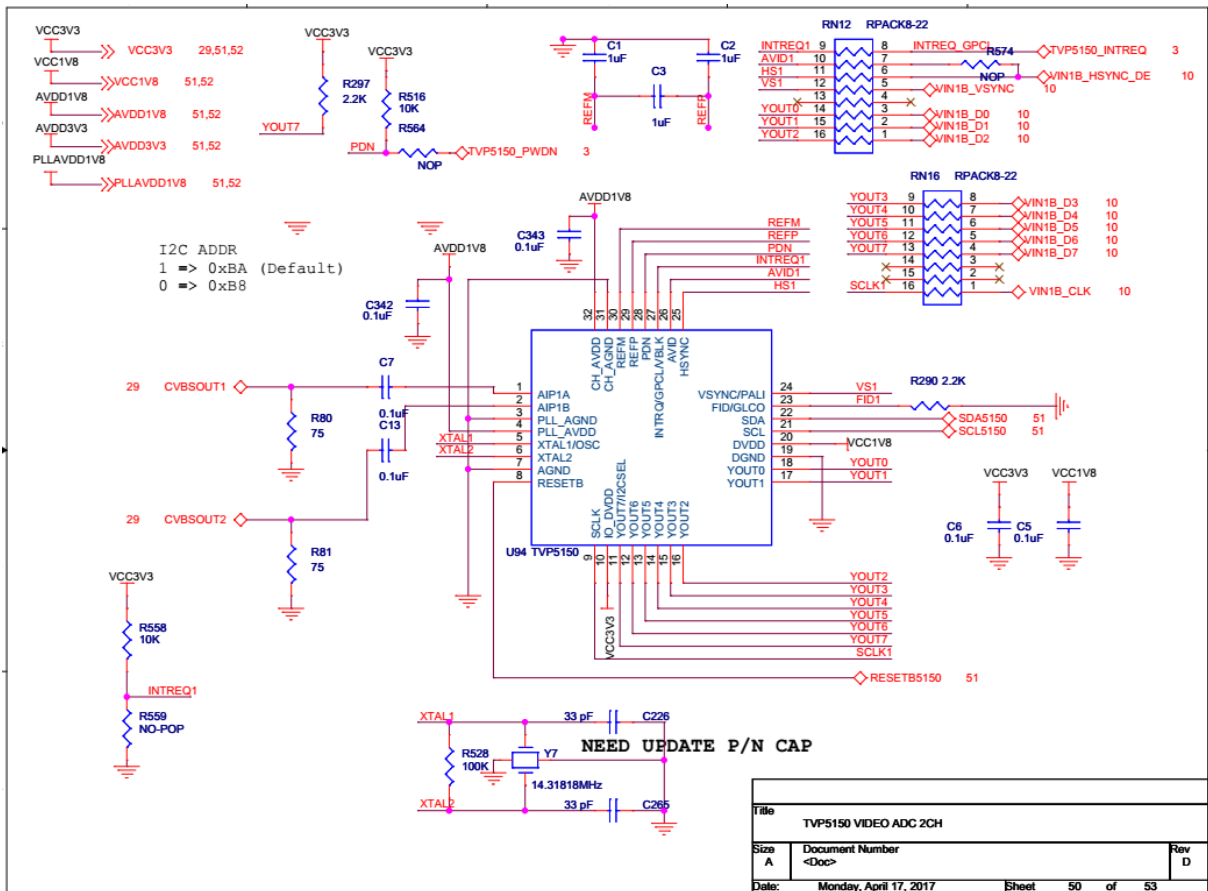
Bảng 2. 2: Mối liên hệ giữa độ phân giải ảnh, tần số quét màn hình và tần số pixel tối đa

Chuẩn	Kích thước (pixel)		Độ phân giải thực (pixel)		Tần số quét màn hình (Hz)	Tần số pixel tối đa (Hz)	Tần số pixel tối đa (MHz)
	W	H	N_H	N_V			
480p	720	480	858	525	25	5,630,625	5.6
					30	6,756,750	6.8
					60	13,513,500	13.5
					100	22,522,500	22.5

D1	720	576	864	625	25	6,750,000	6.8
					30	8,100,000	8.1
					60	16,200,000	16.2
					100	27,000,000	27.0
HD 720p	1280	720	1650	750	25	15,468,750	15.5
					30	18,562,500	18.6
					60	37,125,000	37.1
					100	61,875,000	61.9
HD 1080p	1920	1080	2200	1125	25	30,937,500	30.9
					30	37,125,000	37.1
					60	74,250,000	74.3
					100	123,750,000	123.8

Như vậy, bảng trên cho ta thấy, với mỗi chuẩn (độ phân giải) và tần số quét màn hình khác nhau thì cho ta tần số video tối đa khác nhau. Các IC đọc video đầu vào phổ biến thường hoạt động ở các tần số pixel tối đa phổ biến là: 13.5MHz, 27.0MHz, 75MHz và 125MHz.

Ta xét nguyên lý khối đầu thu tín hiệu video tương tự như sau:



Hình 2. 5: Nguyên lý khối đọc đầu vào tương tự composite

Theo đó, tín hiệu video tương tự từ đầu vào CVBSOUT1&CVBSOUT2 sẽ được đưa vào chip ADC; Tại đây, thực hiện quá trình lấy mẫu. Dữ liệu đầu ra là dữ liệu đã được chuyển đổi kênh màu (mã hóa). Dữ liệu này được đẩy ra trên một bus song song 8/16/24 bit. Tùy thuộc vào cấu tạo của các chip ADC. Trong trường hợp này là 8 bit (YOUT0 đến YOUT7). IC này hoạt động với thạch anh ngoài tần số 14.31818MHz và bộ nhân tần (PLL) sẽ tạo ra tần số pixel 27MHz cung cấp cho quá trình đọc dữ liệu.

1.2. Đầu vào số

Đầu vào video số được bao gồm các chuẩn phổ biến sau:

				
HDMI	SDI	USB 3.0	IP/GigE	Camera Link
- Độ dài cáp: 30m	- Độ dài cáp: 100m	- Độ dài cáp: 10m	- Độ dài cáp: 100m	- Độ dài cáp: 15m
- Bitrate: >1000Mb/s	- Bitrate: 3000Mb/s	- Bitrate: 2400Mb/s	- Bitrate: 1000Mb/s	- Bitrate: >2000Mb/s

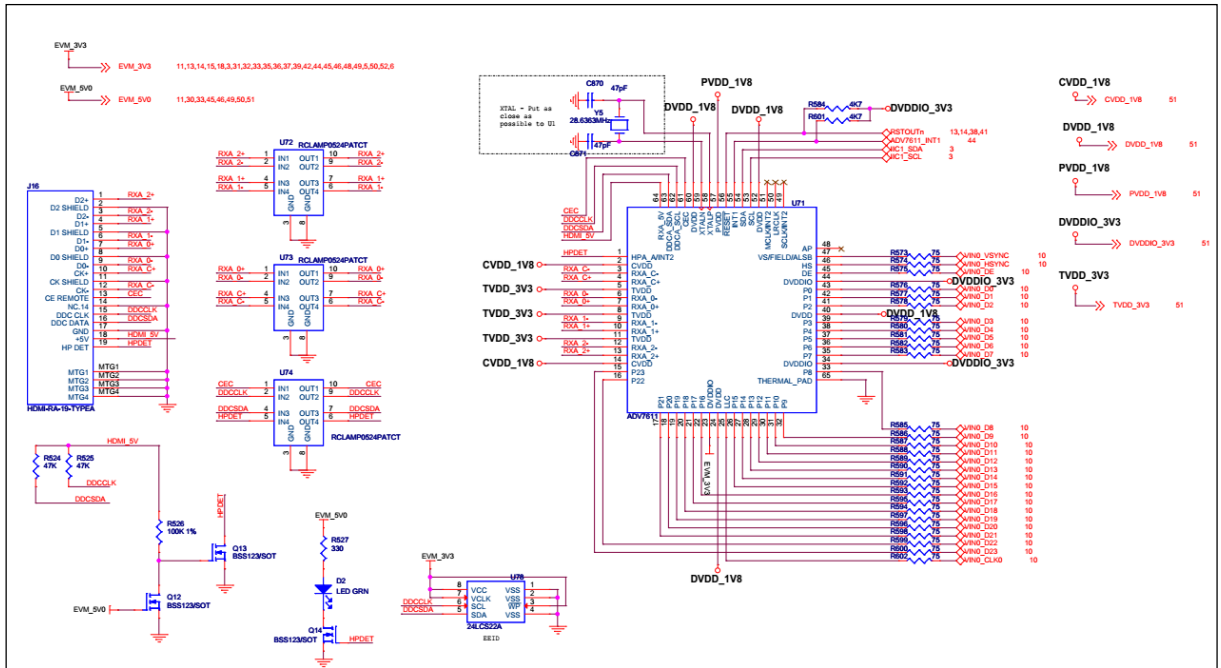
Hình 2. 6: Các chuẩn video số

Bo mạch Apollo17 hỗ trợ các đầu vào: HDMI và IP.

- **Đầu vào HDMI**

Trong hình 3.4 là nguyên lý khối thu tín hiệu HDMI sử dụng chip của hãng Analog Devices (ADV7611).

Tín hiệu HDMI từ đầu vào jack 19 chân (J16) được đưa vào chip nhận (HDMI receiver), thực hiện quá trình lấy mẫu và biến đổi không gian màu. Dữ liệu đầu ra gồm 24bit YCrCb được chuyển sang khối nhận đầu vào video 0 trên DSP (VIN0_D0 đến VIN0_D23).



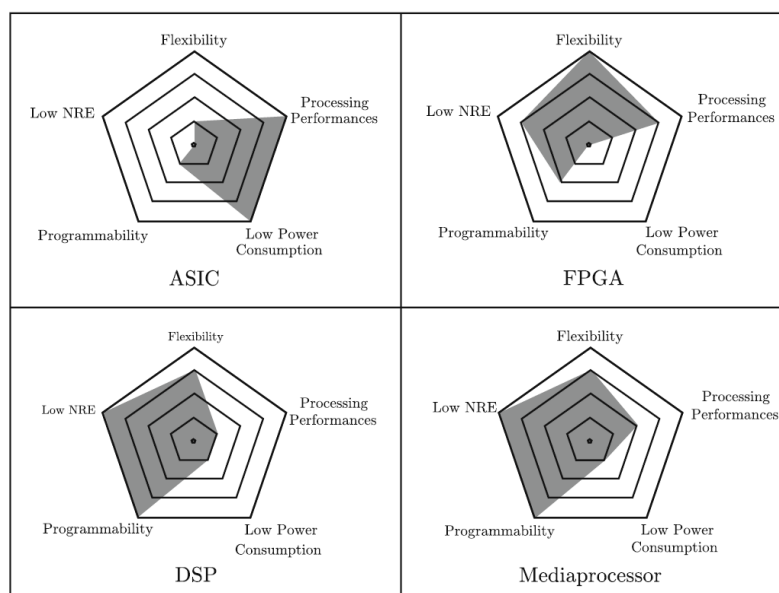
Hình 2. 7: Nguyên lý khối đọc đầu vào số HDMI

2. Khối xử lý

Ngày nay, cùng với sự phát triển của công nghệ bán dẫn và kỹ thuật chế tạo vật liệu, linh kiện bán dẫn. Các công ty sản xuất chip đã cho ra đời rất nhiều các dòng chip mạnh mẽ, hiệu năng cao.

Tuy nhiên, mỗi nền tảng xử lý có một thế mạnh khác nhau. Trong khi FPGA cho phép tính toán đồng thời nhiều dữ liệu độc lập như các phép tính trên ma trận, xử lý lấy mẫu tốc độ cao, song song thì DSP lại cho phép tối ưu các tính toán số học, logic, các vòng lặp. Còn ARM sẽ cho phép việc dễ dàng porting các hệ điều hành nhân Linux lên đó và triển khai các ứng dụng dựa trên hệ điều hành rất thuận tiện và nhanh chóng. Do đó, cách tiếp cận dựa trên mô hình thiết kế kết hợp thế mạnh xử lý của các nền tảng sẽ cho phép giải quyết được một lớp các bài toán xử lý ảnh đòi hỏi không chỉ tốc độ và độ phức tạp tính toán cao mà còn linh hoạt trong điều khiển cơ cấu chấp hành.

Qua nghiên cứu, phân tích tác giả đề xuất lựa chọn khối xử lý kết hợp giữa ARM và DSP. Các phần tính toán phức tạp sẽ được thực thi trên lõi DSP, các phần giao diện, điều khiển sẽ được thực thi trên lõi ARM. Việc lựa chọn 1 chip với các lõi xử lý lai ghép như vậy sẽ giúp tiết kiệm chi phí và giảm kích thước của bo mạch phần cứng.



Hình 2. 8: Biểu đồ đánh giá thế mạnh của các nền tảng

- **DSP:** Phần xử lý chính của thuật toán sẽ được thực hiện ở đây bao gồm các thao tác tính toán phức tạp, các phép tính số học, logic, các vòng lặp và cấu trúc điều khiển rẽ nhánh của thuật toán. Trong quá trình hoạt động, DSP và ARM luôn có sự tương tác trao đổi dữ liệu thông qua các bus giao tiếp và các đường tín hiệu điều khiển thông qua kỹ thuật giao tiếp giữa các quá trình (Inter Process Communication). Kết quả trả về là luồng video đã được xử lý sẽ được đẩy ra ngoài để hiển thị hoặc truyền đi xa.
- **ARM:** Đóng vai trò điều khiển hoạt động chung của toàn hệ thống, cung cấp giao diện ứng dụng, giao diện với các ngoại vi trên môi trường hệ điều hành Linux thân thiện với người dùng.

Bảng 2. 3: So sánh giữa các dòng chip của hãng Texas Instrument.

Tiêu chí so sánh	DM8148	DM8168	DM3730	TMS320C6678
Giá đơn chiếc	48.75 đô la/1 chip	75.00 đô la /1 chip	30.75 đô la /1 chip	661.26 đô la /1 chip
Ứng dụng (Applications)	Video Security Thin Client Video Conferencing Video Phones)	Video Security Thin Client Video Conferencing Video Phones)	Audio Automotive Communications and Telecom Computers and Peripherals Consumer Electronics Energy Industrial Medical Security	Automation and Process Avionics and Defense Communications and Telecom Consumer Electronics Industrial Medical Security Space Video and Imaging

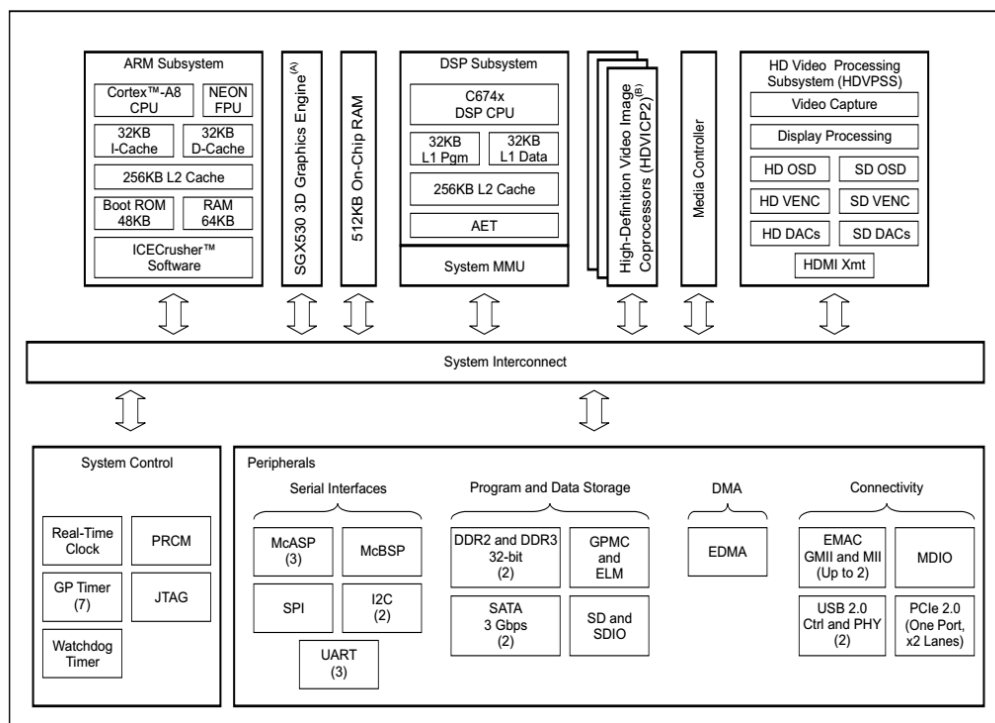
Hệ điều hành (Operating Systems)	Linux DSP/BIOS	Linux Android DSP/BIOS	Android DSP/BIOS Neutrino ntegrity Windows Embedded CE Linux VXWorks	SYS/BIOS Linux VxWorks Integrity
Lõi xử lý ARM	1 ARM Cortex-A8	1 ARM Cortex-A8	1 ARM Cortex-A8	4 ARM Cortex-A15
Tần số lõi xử lý ARM (Max.)	1000	1200	800	1400
Lõi xử lý DSP	1 C674x	1 C674x	1 C64x	8 C66x
Tần số lõi xử lý DSP(Max.)	750	1000	800	1200
Bộ tăng tốc video (Video Acceleration)	1 HDVICP	3 HDVICPs	x	Packet Accelerator Security Accelerator
Ngõ vào/ra video (Video Port Configurable)	2 Output 2 Input 2 SD DACs 1 HDMI TX	1 HDMI TX 2 Input 2 Output 3 HD DACs 4 SD DACs	1 Dedicated Input 1 Dedicated Output	
USB	2	2	4	1
PCI/PCIe	PCIe	1 2-lane	x	2 PCIe Gen2
EMAC	10/100/1000	2x 10/100/1000	10/100	4-port 1Gbe Switch
DRAM	LPDDR DDR2 DDR3	DDR2 DDR3	LPDDR	DDR3 DDR3L
SPI	4	1	4	3
I2C	4	2	4	3
UART (SCI)	6	3	4	2
Bộ nhớ nội (On-Chip L2 Cache)	512 KB (ARM) 256 KB (DSP)	256 KB (ARM Cortex-A8) 256 KB (DSP)	256 KB (ARM Cortex-A8)	6144 KB
Dải nhiệt độ hoạt động (Operating Temperature Range)	0 ÷ 90 (C)* -40 ÷ 125 (I)**	0 ÷ 90 (C) -40 ÷ 125 (I)	0 ÷ 90 (C) -40 ÷ 125 (I)	0 ÷ 90 (C) -40 ÷ 125 (I)
Công cụ phát triển	Linux EZ Software Development Kit (EZSDK) for DaVinci(TM) DM814x and	Linux EZ Software Development Kit (EZSDK) for DaVinci(TM) DM814x and	Linux Digital Video Software Development Kit (DVSDK) for DM3730/3725 Digital Media Processors	Multicore Video Infrastructure Demo for Multicore Software Development Kit (MCSDK)

	DM816x Video Processors	DM816x Video Processors		
KIT tham khảo	http://z3technology.com/upload/DOC-MKT-0015-03_Z3-DM8148-RPS_Data_Sheet.pdf	http://z3technology.com/upload/DOC-MKT-0001-05_Z3-DM8169-VI-RPS_Data_Sheet.pdf	https://www.isee.biz/products/igep-processor-boards/igepv2-dm3730	http://www.ti.com/tool/evmk2h

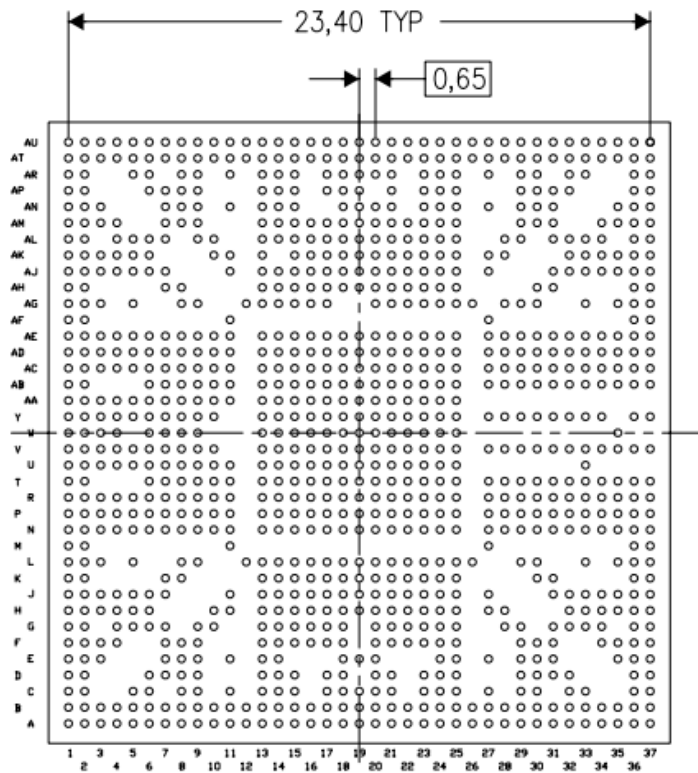
Ghi chú: (*) C = Commercial – chuẩn thương mại; (**) I = Industrial – chuẩn công nghiệp.

Chip xử lý được chọn là TMS320DM8168 của hãng Texas Instrument – Mỹ với giá cả phù hợp và các thông số kỹ thuật nổi trội như trong bảng so sánh trên.

Hình 3.6 là kiến trúc phần cứng của chip TMS320DM8168 với tên gọi kiến trúc Davinci.



Hình 2. 9: Kiến trúc chip xử lý TMS320DM8168



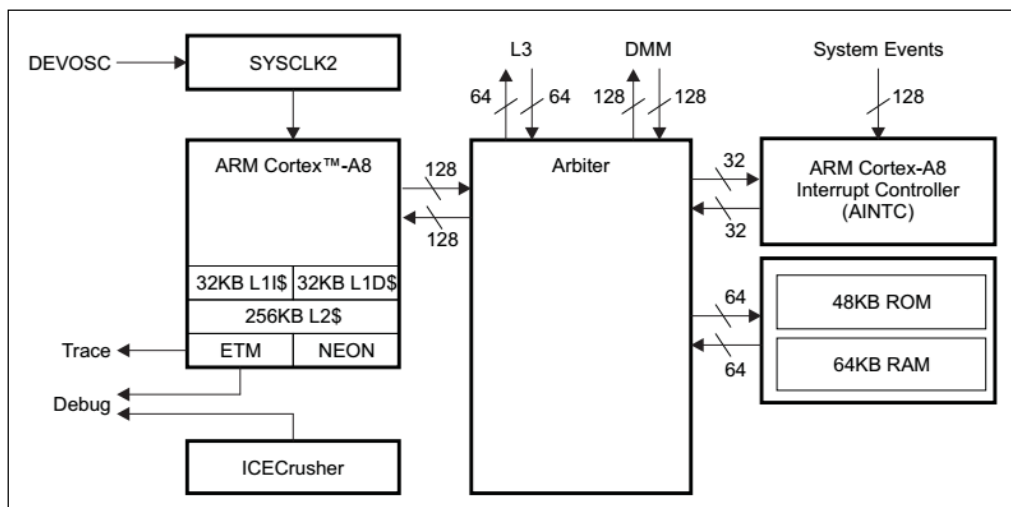
Hình 2. 10: Bố trí chân của chip TMS320DM8168

(gồm 1031 chân dạng cầu, khoảng cách các chân 0.65mm)

Trong đó,

- **Lõi xử lý ARM:**

Lõi xử lý ARM được thiết kế sử dụng kiến trúc ARM Cortex-A8. Lõi xử lý này có thể được dùng để điều khiển việc trao đổi dữ liệu giữa ARM-DSP, điều khiển hoạt động của các ngoại vi và bộ nhớ ngoài (bộ nhớ DDR3).



Hình 2. 11: Kiến trúc lõi xử lý ARM

Lõi xử lý ARM bao gồm các tài nguyên sau:

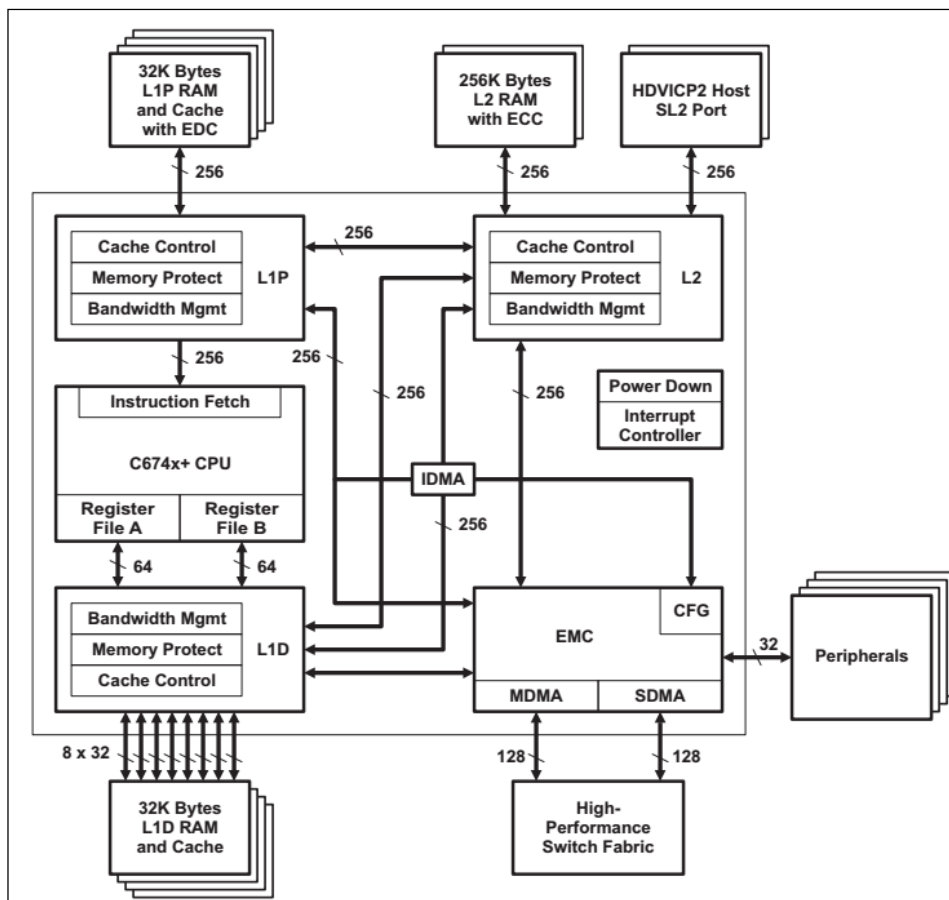
- Nhân xử lý ARM Cortex-A8 kiến trúc RISC.

- Khôi theo dõi các module (ETM)
- Bộ điều khiển ngắt (AINTC)
- Bộ nhớ đệm nội 64KB
- Bộ nhớ nội chỉ đọc 48KB.

• **Lõi xử lý DSP:**

Lõi xử lý DSP bao gồm các tài nguyên:

- Khối xử lý tính toán C674x
- 32KB bộ nhớ chương trình L1P với mã phát hiện lỗi (EDC)
- 32KB bộ nhớ dữ liệu L1D
- 256KB bộ nhớ đệm L2 với mã sửa lỗi (ECC)
- Kết nối trực tiếp đến khối HDVICP2



Hình 2. 12: Kiến trúc lõi xử lý DSP

Bộ não của lõi xử lý DSP là khối xử lý trung tâm C674x. CPU này bao gồm 8 đơn vị chức năng, hai tệp thanh ghi, hai đường dẫn dữ liệu như thể hiện trong hình 3.10

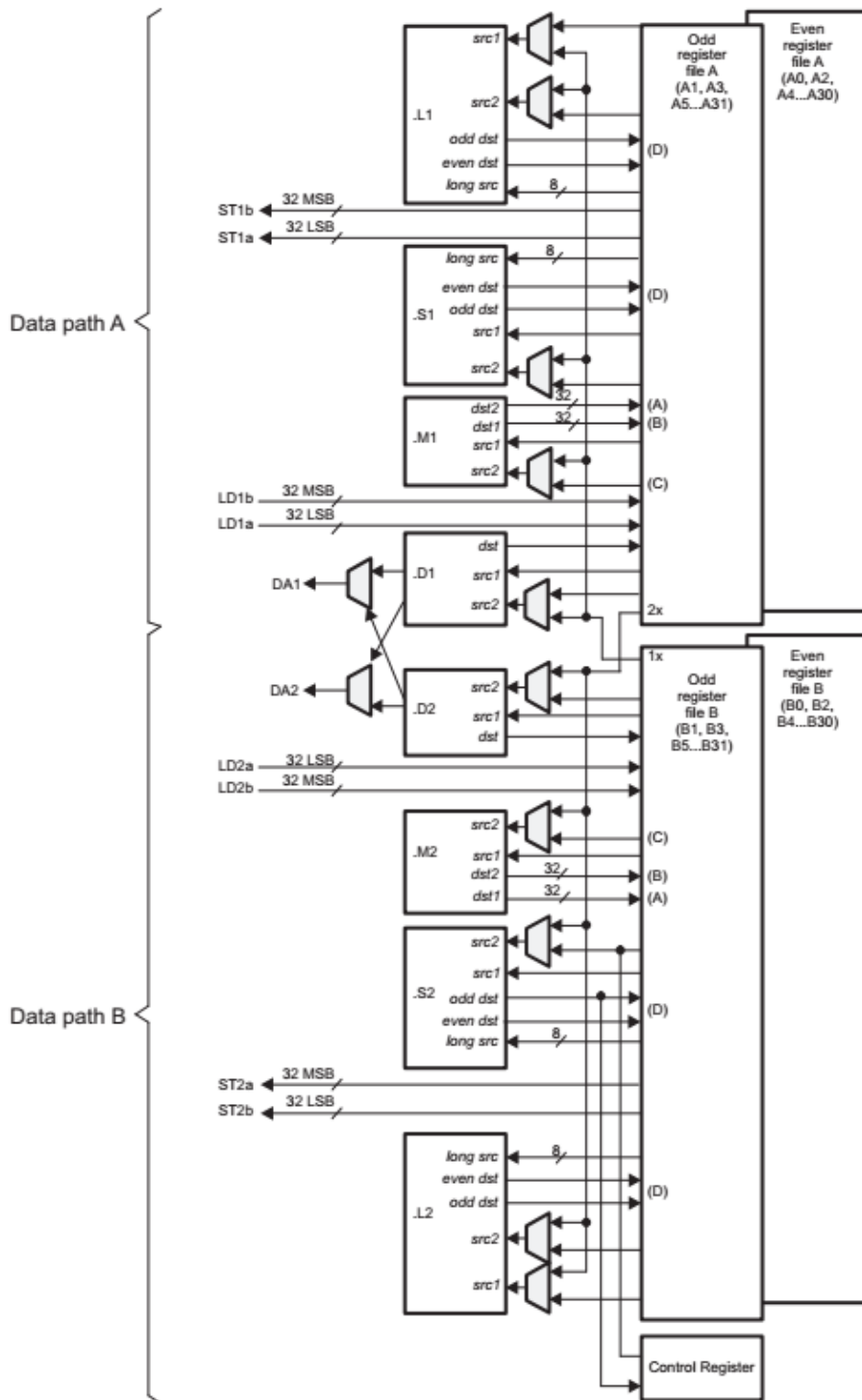
Hai tệp thanh ghi chức năng chung (A và B) mỗi tệp chứa 32 thanh ghi 32 bit. Tạo nên tổng 64 thanh ghi chứa dữ liệu hoặc con trỏ dữ liệu trong quá trình xử lý tính toán của CPU.

Tám đơn vị chức năng gồm (.M1, .L1, .D1, .S1, .M2, .L2, .D2 và .S2) đều có khả năng thực hiện một chỉ lệnh trong một chu kỳ xung nhịp (clock cycle).

Đơn vị chức năng .M thực hiện toàn bộ các phép tính nhân. Các đơn vị .S và .L

thực hiện một tập các phép tính số học, lô-gic và rẽ nhánh. Đơn vị chức năng .D chủ yếu thực hiện việc tải dữ liệu từ bộ nhớ đến các thanh ghi và lưu kết quả từ thanh ghi vào bộ nhớ.

Điều đặc biệt là mỗi đơn vị .M của CPU C674x có khả năng thực hiện trong 1 chu kỳ xung nhịp: một phép nhân 32x32 bit, một phép nhân 16x32 bit, hai phép nhân 16x16 bit, 4 phép nhân 8x8 bit... Ưu thế này được tận dụng để tối ưu các thuật toán xử lý ảnh giúp cho việc tính toán giảm đi rất nhiều lần.



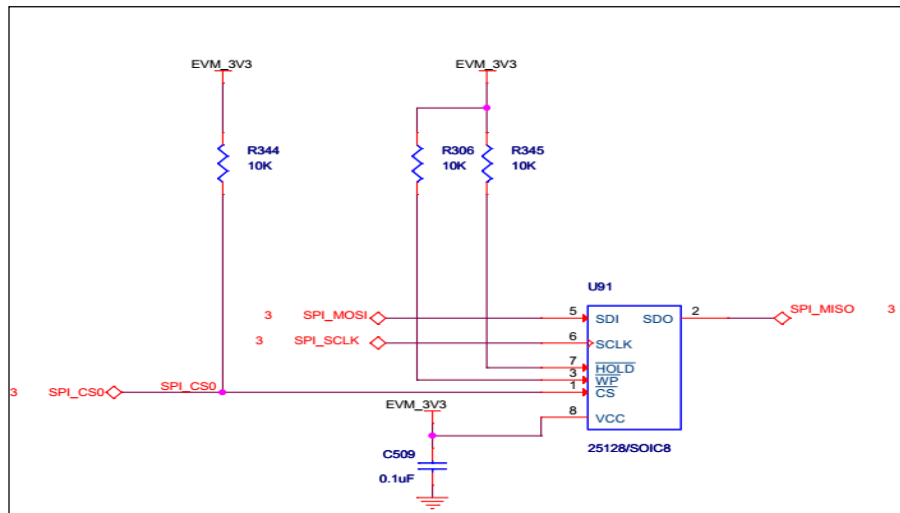
Hình 2. 13: Kiến trúc CPU C674x

3. Bộ nhớ

Bộ nhớ sử dụng trong thiết kế của Apollo17 gồm 2 loại sau:

3.1. Bộ nhớ lưu trữ

Bộ nhớ lưu trữ bao gồm IC nhớ Flash cấu trúc SPI được dùng để lưu trữ chương trình u-boot và thẻ nhớ để lưu trữ nhân linux đã được biên dịch cộng với các file hệ thống (root file system).



Hình 2. 14: Nguyên lý kết nối đến bộ nhớ SPI Flash

Lỗi xử lý ARM sẽ giao tiếp với bộ nhớ flash thông qua giao thức SPI. Quá trình khởi động sẽ gồm 2 bước:

- Bước 1: Hệ thống sẽ gọi đến chương trình u-boot trên bộ nhớ SPI. Cấu hình các lỗi xử lý và các ngoại vi cần thiết. Bước này gọi là “first stage boot”
- Bước 2: Sau khi đã khởi tạo thành công, hệ thống gọi đến nhân linux. Lúc này nhân linux được copy vào bộ nhớ đệm DDR3 để thực thi và tiếp tục gọi đến các driver còn lại cũng như giao diện hệ điều hành. Bước này gọi là “second stage boot”.

Ngoài ra, cũng có thể sử dụng 1 thẻ nhớ microSD duy nhất để lưu trữ cả chương trình u-boot và nhân Linux cộng với các file hệ thống.

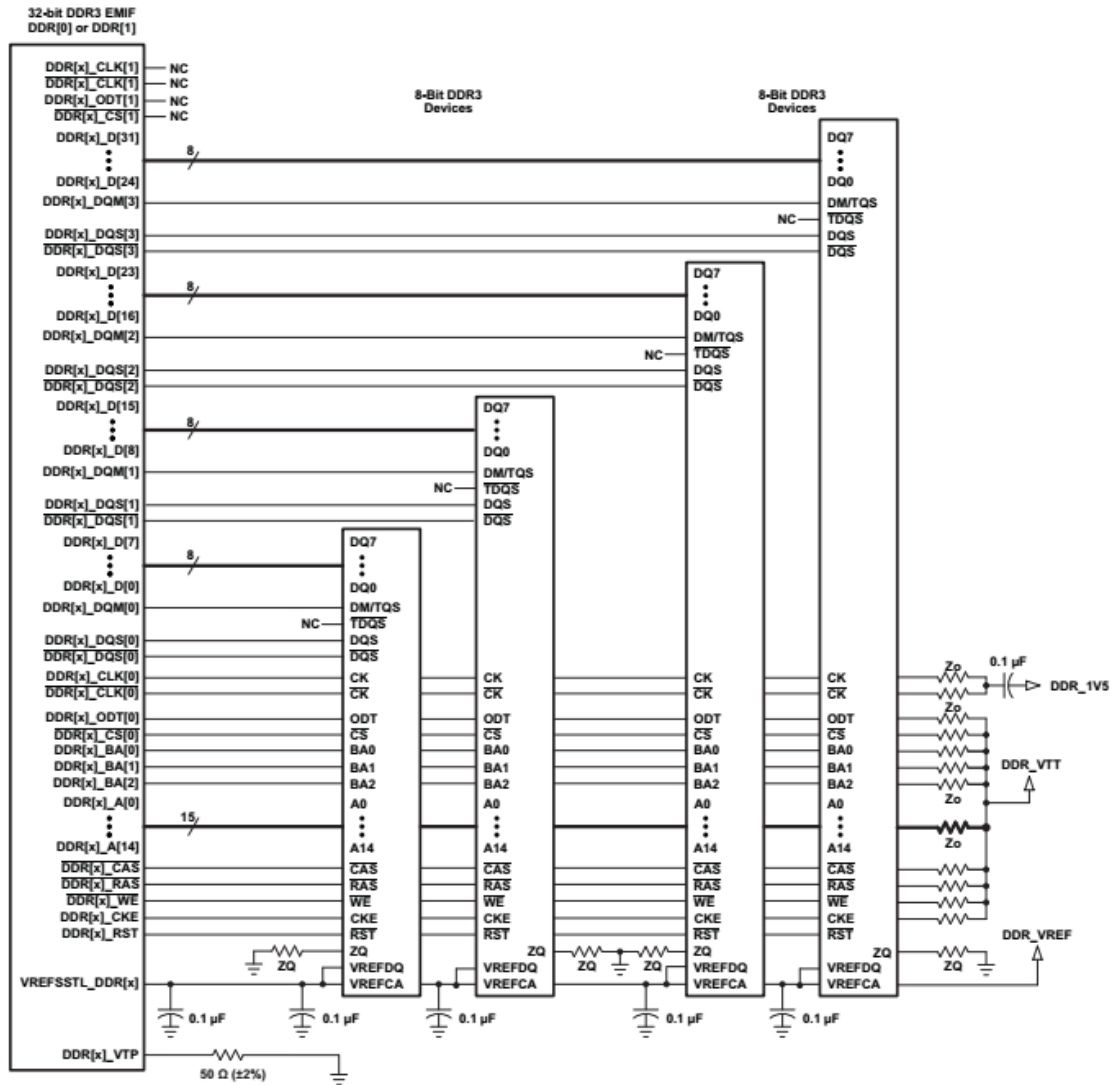
Quá trình khởi động cũng vẫn gồm 2 bước như trên.

Thời gian khởi động tùy thuộc vào kích thước của các file hệ thống và số driver được cài đặt. Thông thường thời gian khởi động từ 15 đến 30 giây.

3.2. Bộ nhớ đệm

Bộ nhớ đệm được sử dụng trong thiết kế này là bộ nhớ DDR3 (Double Data Rate Type 3). Là một loại bộ nhớ truy cập ngẫu nhiên động, có tốc độ truy cập dữ liệu cao.

Chip TMS320DM8168 bao gồm 2 khối điều khiển bộ nhớ đệm với độ rộng dữ liệu trao đổi tối đa là 32 bit.



Hình 2. 15: Mô hình kết nối DDR3 8 bit.

Như vậy, nếu dùng DDR3 loại 8 bit thì phải cần kết nối 4 DDR3 vào 1 bộ điều khiển EMIF trên chip xử lý.

Trong trường hợp dùng DDR3 loại 16 bit thì ta chỉ cần kết nối 2 DDR3 cho một bộ điều khiển.

Ngoài ra, việc thiết kế layout khối DDR3 khá phức tạp và cần tuân thủ những nguyên tắc khắt khe về độ dài các đường tín hiệu (xung nhịp, dữ liệu và địa chỉ), khoảng cách giữa các IC và việc bố trí các tụ lọc.

Tham khảo các kỹ thuật layout cho DDR3 trong tài liệu [11] - “DDR3 point-to-point design support” của Micron.

4. Đầu ra video

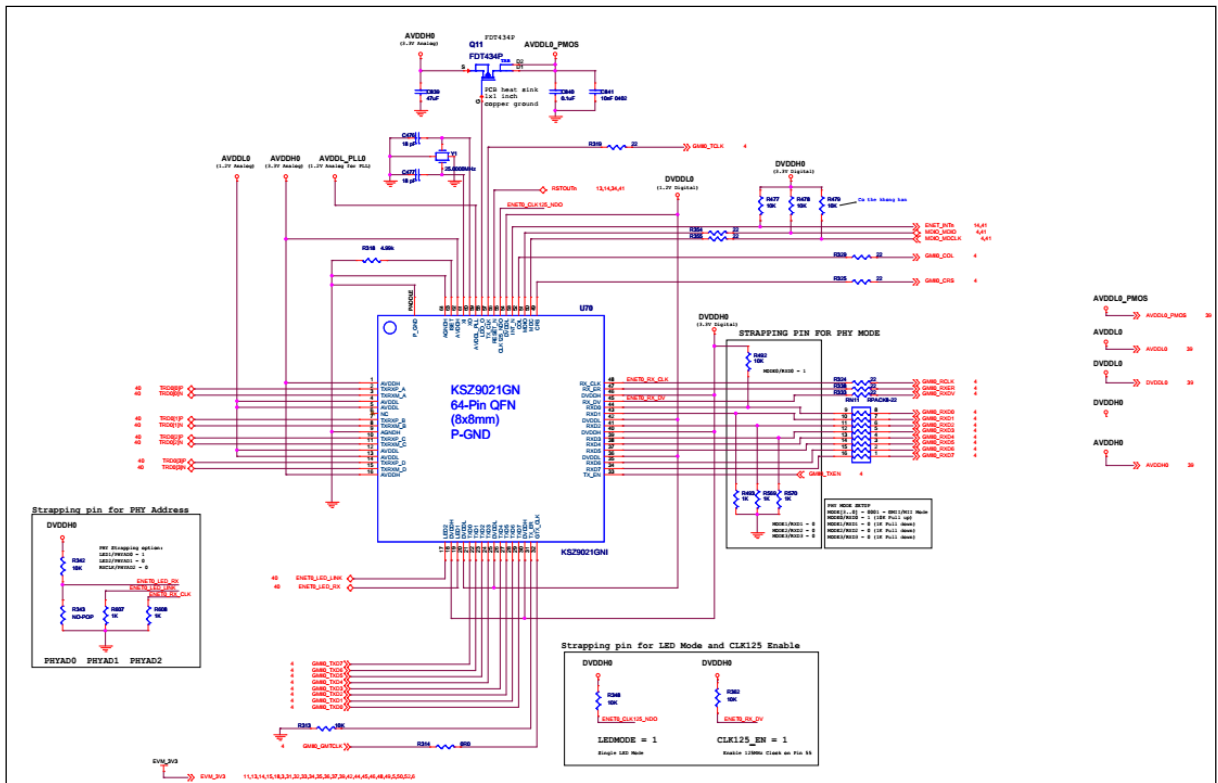
Video sau khi xử lý được truyền ra ngoài theo 2 cách
 Một là, hiển thị ra màn hình thông qua giao diện HDMI
 Hai là, được nén và truyền ra ngoài thông qua giao diện IP.

- **Đầu ra HDMI**

Chip TMS320DM8168 hỗ trợ 1 đầu ra HDMI trực tiếp. Do vậy, việc xuất dữ liệu ra màn hình HDMI khá đơn giản. Trong thiết kế chỉ cần sử dụng thêm một IC đệm mức tín hiệu kết nối với jack HDMI chuẩn là đạt yêu cầu.

- **Đầu ra IP**

Khối truyền thông Ethernet trên chip TMS320DM8168 hỗ trợ tốc độ truyền tối đa 1Gbps.



Hình 2. 16: Nguyên lý kết nối IC giao tiếp Ethernet

Thiết kế khối đầu ra IP sử dụng 1 IC chuyên dụng giúp truyền tải tín hiệu mạng ở tầng vật lý.

5. Khối cấp nguồn

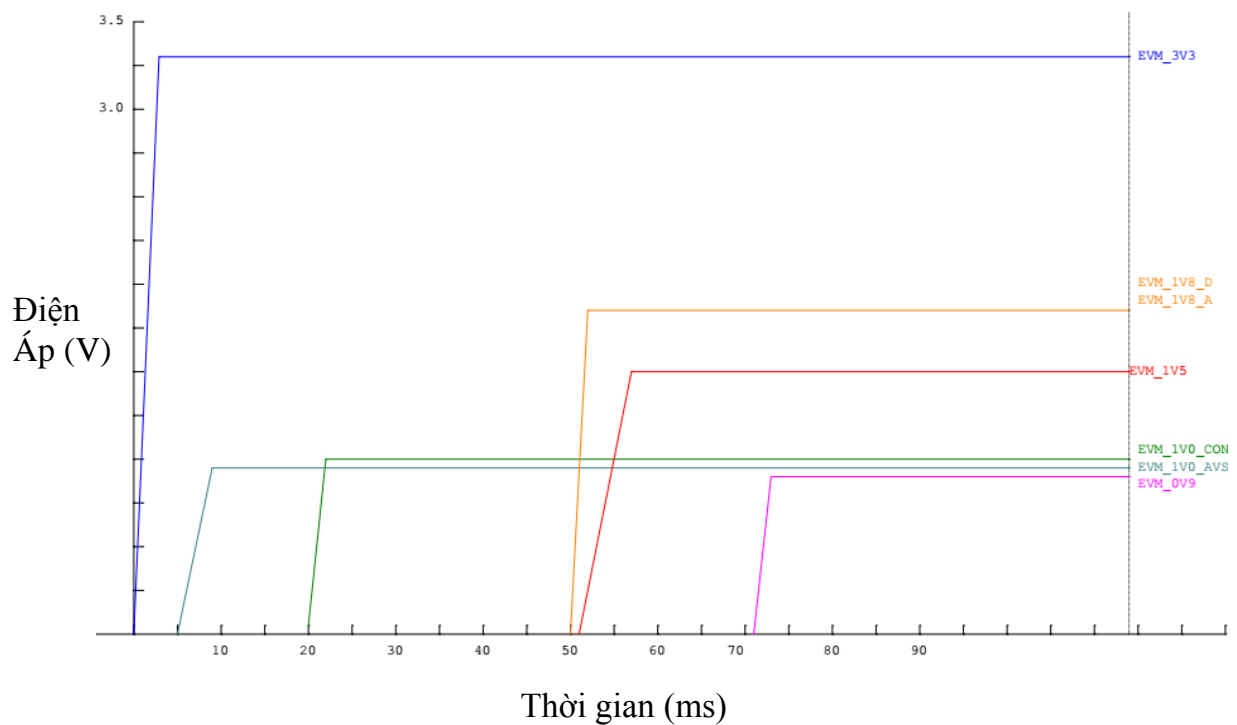
Nguồn cấp với đầu vào dải rộng 5V ~ 36V@3A cung cấp nguồn tới các mạch nguồn thứ cấp 3,3V, 1,8V, 1,2V, 1,0V cấp cho các chipset trên mạch.

Số lượng các nguồn thứ cấp được liệt kê trong bảng sau:

Bảng 2. 4: Danh sách nguồn cấp cho chip DSP

TT	Điện áp (Volt)	Sử dụng
1	12	Nguồn đầu vào
1	5	Nguồn nuôi thứ cấp tổng. Cung cấp cho các IC nguồn thứ cấp thấp hơn
2	3,3	Cung cấp cho DSP, khối HDMI, cung cấp cho các IC nguồn thứ cấp thấp hơn
3	1,0	Cung cấp cho DSP
4	1,5	Cung cấp cho DSP, DDR3
5	0,9	Cung cấp cho DDR3
6	1,0	Cung cấp cho DSP
7	1,8	Cung cấp cho DSP

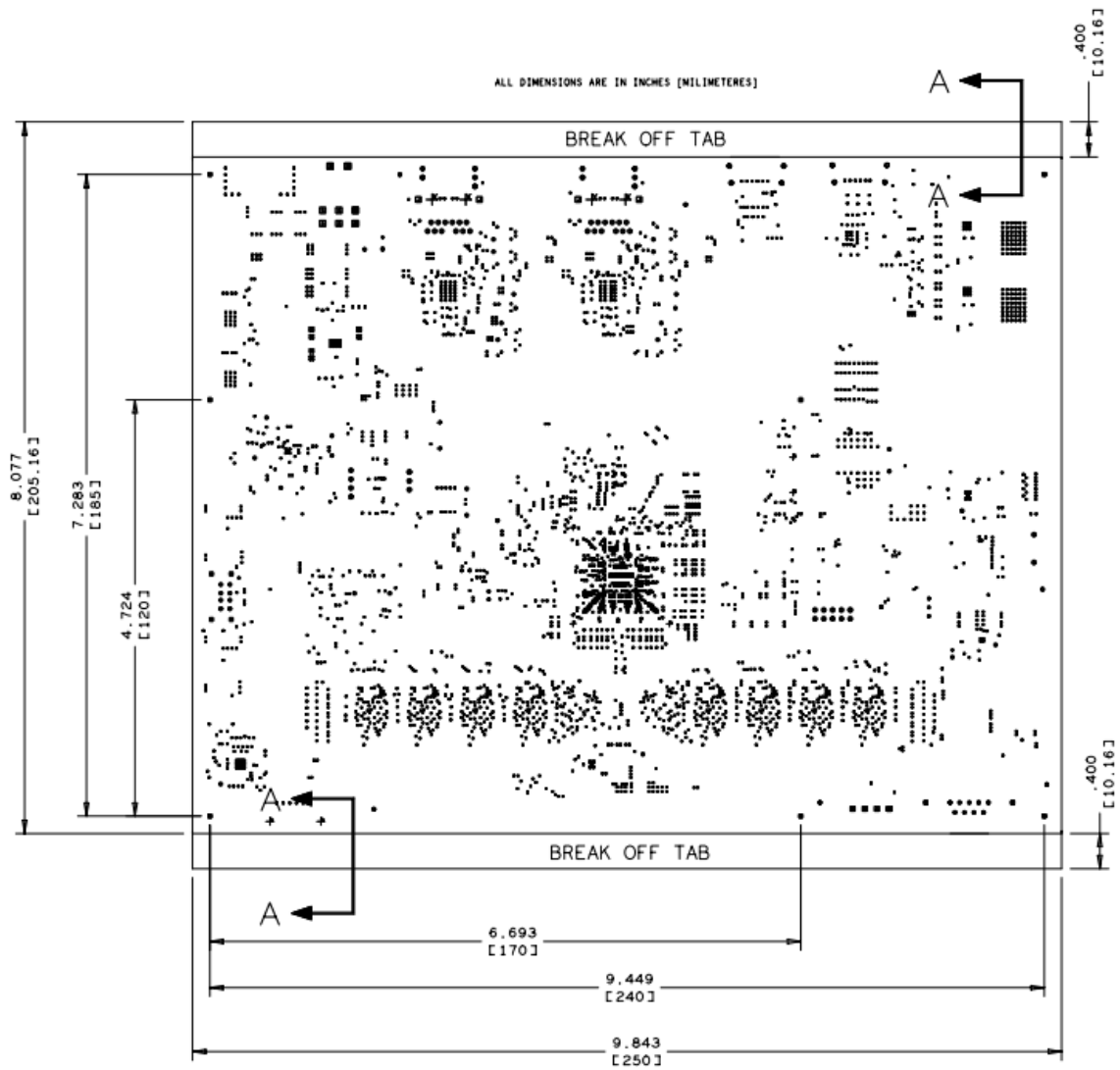
Thứ tự khởi động của các nguồn cấp dưới 3.3V phải tuân thủ như hình sau:



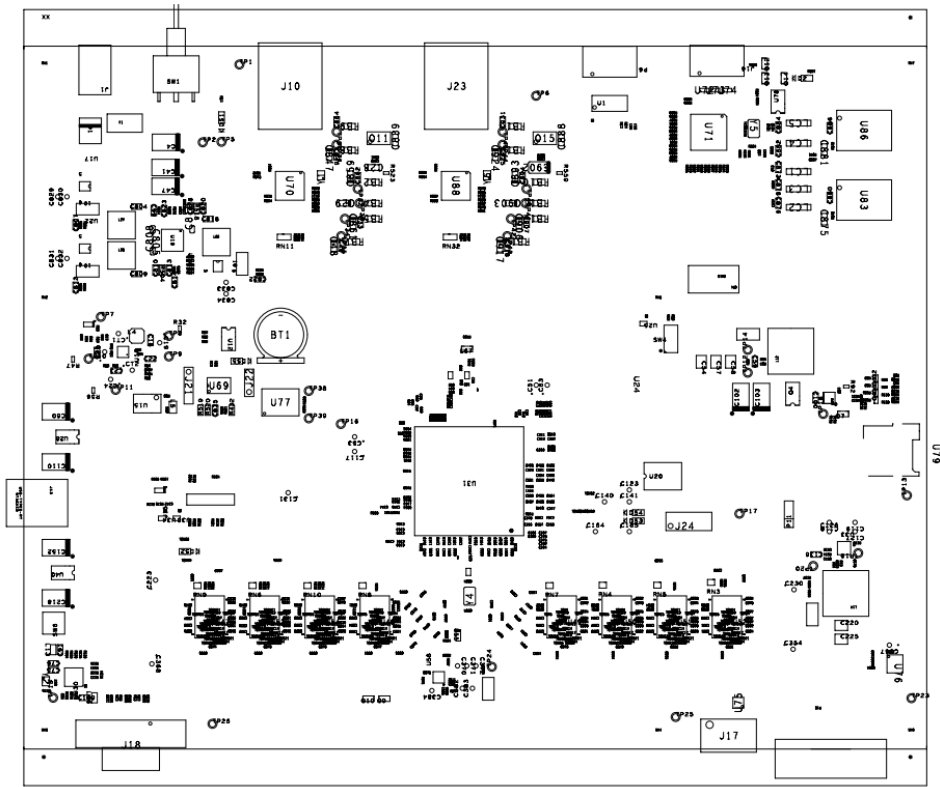
Hình 2. 17: Thứ tự khởi động của các nguồn cấp cho DSP

6. Gia công chế tạo

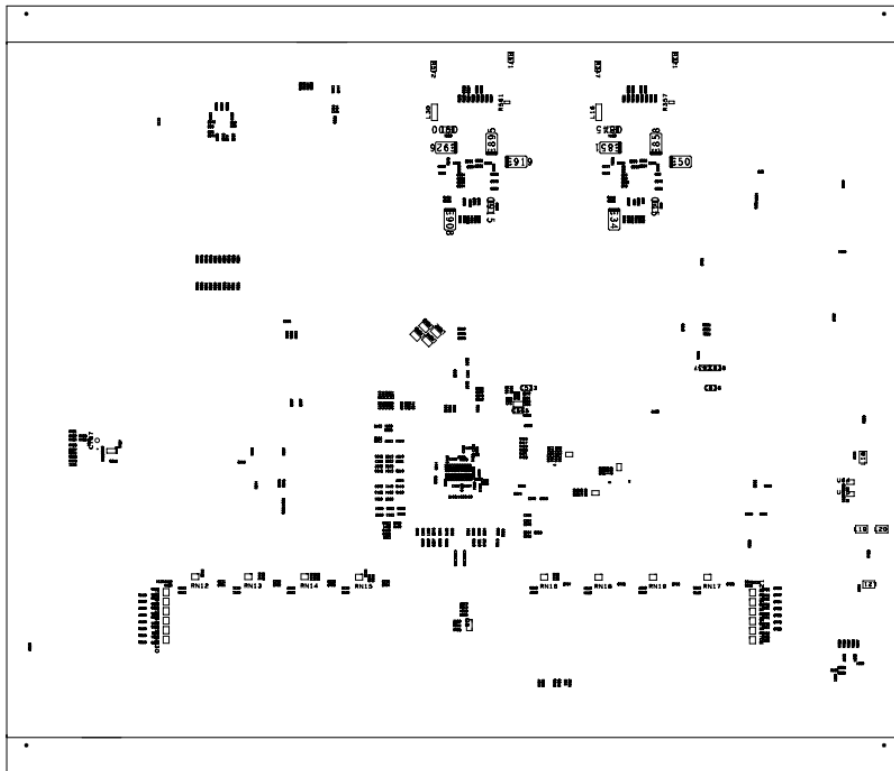
Sản phẩm đã hoàn thiện phiên bản 1.0 và được gia công chế tạo thử nghiệm tại một nhà máy ở Hàn Quốc.



Hình 2. 18: Bản vẽ gia công bo mạch thử nghiệm



Hình 2. 19: Bản vẽ lắp ráp linh kiện mặt trên



Hình 2. 20: Bản vẽ lắp ráp linh kiện mặt dưới

CHƯƠNG 3: ĐÁNH GIÁ HỆ THỐNG VỚI THUẬT TOÁN MẪU

- Giới thiệu thuật toán
- Triển khai thuật toán và đánh giá

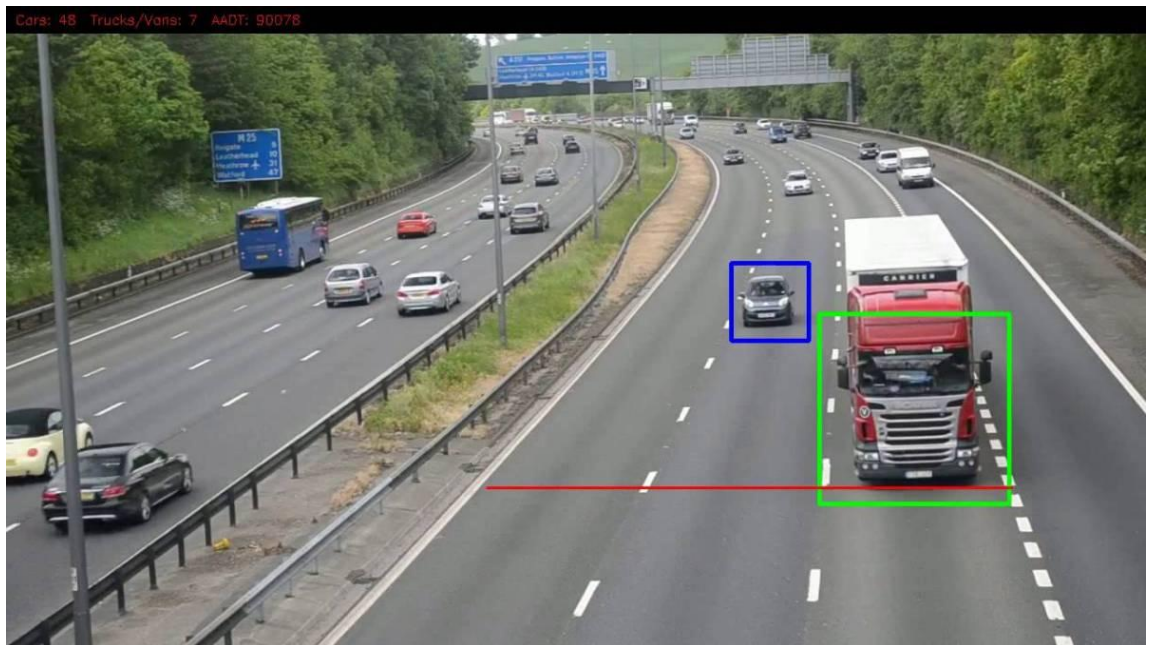
Như vậy, sau khi hoàn thành chế thử bo mạch mẫu. Chúng ta bắt tay vào quá trình bring-up bo mạch (*tham khảo mục [12] - Bộ công cụ hỗ trợ phát triển phần mềm Linux EZSDK*), thiết kế phần mềm và thử nghiệm thuật toán mẫu để đánh giá khả năng xử lý của thiết bị. Thuật toán được lựa chọn để đánh giá là thuật toán phát hiện mục tiêu chuyển động. Đây là một bài toán kinh điển trong xử lý ảnh và có nhiều ứng dụng trong thực tế, đặc biệt là các bài toán giám sát an ninh.

1. Giới thiệu thuật toán

1.1. Đặt vấn đề

Phát hiện đối tượng chuyển động trong chuỗi hình ảnh là một trong những bước đầu tiên của bài toán giám sát đối tượng, nhằm đưa ra các thông tin về vùng chứa đối tượng chuyển động, số lượng và mật độ đối tượng.

Phát hiện chuyển động cung cấp đầu vào cho quá trình theo dõi đối tượng.



Hình 3. 1: Minh họa phát hiện chuyển động

Trong đề tài này, xem xét ngữ cảnh video sử dụng thử nghiệm là các video được quay từ các camera lắp cố định theo dõi một vùng khung cảnh tĩnh, tức ảnh nền trong video thu được là ảnh nền tĩnh không thay đổi. Khi đó các đối tượng chuyển động có thể phát hiện bằng cách so sánh frame hiện tại với 1 ảnh nền (background) để tách nền

ra khỏi ảnh hiện tại. Phương pháp này gọi chung là background subtraction. Sau khi tách nền, một loạt các phép hậu xử lý sẽ được áp dụng để khoanh vùng đối tượng.

Các thuật toán phát hiện chuyển động khi đưa xuống triển khai trên nền tảng nhúng được đánh giá dựa trên các yếu tố sau:

- Độ phức tạp về thời gian tính toán
- Yêu cầu về bộ nhớ để lưu trữ các tham số thuật toán
- Hiệu quả phát hiện chuyển động

1.2. Mô tả thuật toán

Như đã giới thiệu ở phần trước, đối với video có ảnh nền tĩnh (static scene), các đối tượng chuyển động có thể phát hiện bằng cách so sánh frame hiện tại với 1 ảnh nền (background). Mô hình chung cho thuật toán Background Subtraction như sau:



Hình 3. 2: Mô hình của các thuật toán Background subtraction

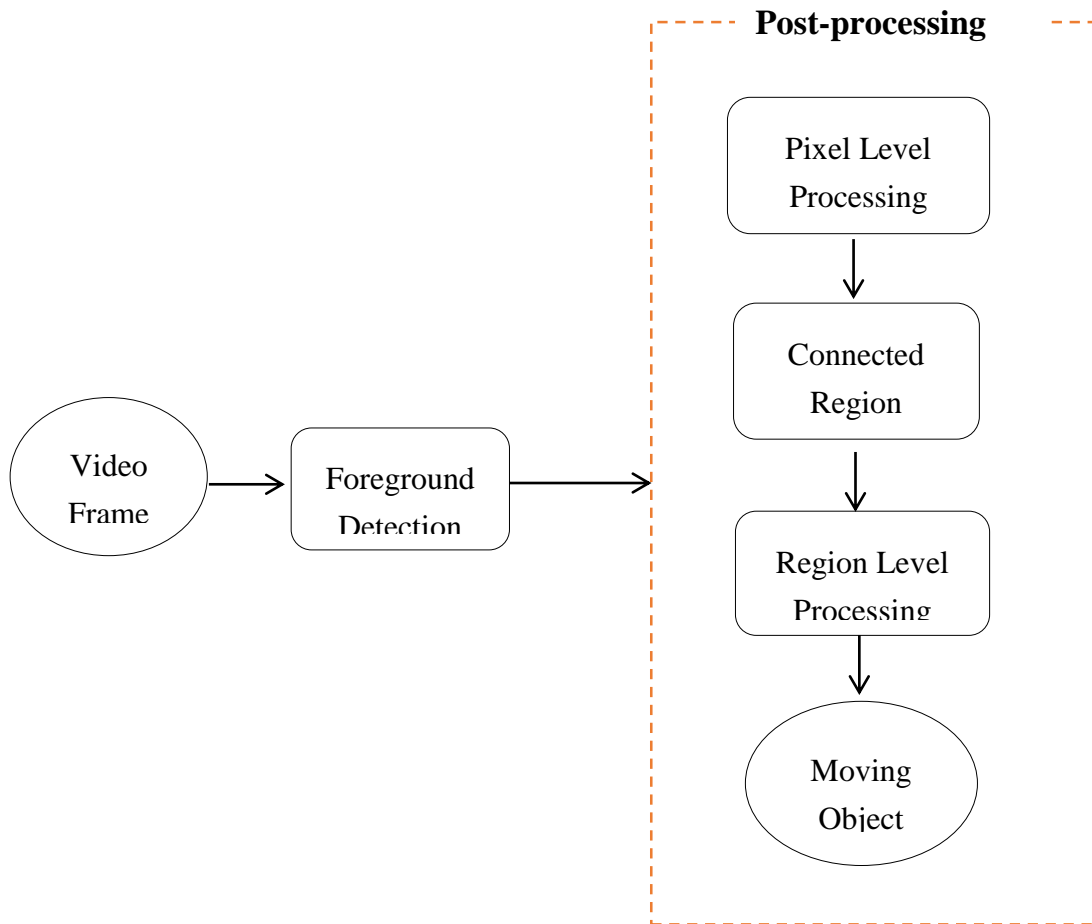
Phép tách nền để tìm ảnh foreground được thực hiện nhờ một background model. Background model được xây dựng, cập nhật và sử dụng trong từng frame ảnh. Tại mỗi thời điểm, frame ảnh hiện tại được đem đối sánh với mô hình background, pixel nào có giá trị mức xám chênh lệch vượt quá 1 ngưỡng T theo một độ đo thì được coi là foreground pixel. Sau đó một loạt phép hậu xử lý (post-processing) được áp dụng để hiệu chỉnh lại các foreground pixel và đưa ra tọa độ vùng chứa đối tượng chuyển động thực sự trong ảnh.

Một số mô hình background phổ biến :

- Difference frame : Tại frame hiện tại, sử dụng frame trước đó như là background
- Adaptive Median Filtering (AMF) (hoặc Running Average) : Background được tính dựa trên trung bình có trọng số của các frame trước đó (các frame gần thời điểm hiện tại thì có trọng số cao hơn).
- Mixture Of Gaussian (MOG) : Giá trị mỗi pixel được mô hình hóa bởi K phân bố Gaussian, cho phép giải quyết trường hợp tại một pixel có nhiều giá trị được coi là background.
- Running Gaussian Average (RGA): Trường hợp riêng của MOG, chỉ sử dụng 1 phân phối Gaussian cho mỗi pixel.
- Visual Background Extractor (VIBE) : Không sử dụng mô hình thống kê để đại diện cho phân bố của pixel mà sử dụng tập các giá trị pixel ở N frame trước đó. Việc cập nhật mô hình dựa trên phương pháp chọn lọc ngẫu nhiên.

Thuật toán Runing Gaussian Average coi mỗi pixel của nền có giá trị mức xám biến động theo một phân phối Gaussian. Khi đối tượng chuyển động đi qua nền, giá trị mức xám của pixel sẽ thay đổi đột ngột, nằm ngoài phân bố Gaussian của pixel. Thuật toán RGA đạt được độ chính xác tốt với nền ít động, với tốc độ tính toán cao, dễ triển khai trên nền tảng nhúng.

1.3. Nội dung và lưu đồ xử lý thuật toán



Hình 3. 3: Sơ đồ thuật toán phát hiện chuyển động

❖ Module xử lý chính (Foreground Detection)

Mỗi pixel tại 1 vị trí nào đó được đặc trưng bởi một phân phối Gaussian sau:

$$P(x_t) = \eta(x_t - \mu_t, \Sigma_t) \quad (3.1)$$

Trong đó:

$P(x_t)$: là xác suất tại thời điểm t , quan sát được pixel này có giá trị mức xám là x_t

μ_t : là giá trị trung bình của phân phối Gaussian tại thời điểm t

σ_t : là độ lệch chuẩn của phân phối Gaussian tại thời điểm t

$\Sigma_t = \sigma_t^2 I$: là ma trận hiệp phương sai của phân phối Gaussian tại thời điểm t

Tại thời điểm $t + 1$, khi pixel có một giá trị mức xám mới, giá trị này sẽ được đối sánh xem có thuộc về phân bố Gaussian không dựa trên khoảng cách Euclidean giữa giá trị pixel hiện tại và giá trị trung bình. Nếu pixel thuộc phân phối Gaussian (matching) thì pixel này được coi là background, ngược lại pixel được coi là foreground. Dưới đây là các bước xây dựng mô hình thống kê của thuật toán.

- **Khởi tạo mô hình:**

Sử dụng N frame đầu tiên trong chuỗi ảnh để khởi tạo mô hình Gaussian cho mỗi pixel. Với mỗi pixel, ta sẽ khởi tạo:

$$\mu_0 = \frac{1}{N} \sum_{i=1}^N x_i \quad \text{và} \quad \sigma_0^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_0)^2 \quad (3.2)$$

x_i : là giá trị mức xám của pixel này trong frame thứ i của N frame khởi tạo

Khi $N = 1$, ta có $\mu_0 = x_0$ và $\sigma_0 = m$ được chọn sẵn dựa trên đánh giá về sai khác giữa pixel background và foreground (thường lấy m từ 5 đến 20).

- **Cập nhật mô hình:**

Giả sử ta có tham số μ_{t-1}, σ_{t-1} của phân phối Gaussian tại thời điểm $t-1$. Tại thời điểm t hiện tại, pixel có giá trị mức xám mới là x_t . Các tham số được cập nhật như sau:

$$\begin{aligned} \mu_t &= \alpha x_t + (1 - \alpha) \mu_{t-1} \\ \sigma_t^2 &= \alpha d^2 + (1 - \alpha) \sigma_{t-1}^2 \\ d &= |x_t - \mu_t| \end{aligned} \quad (3.3)$$

Trong đó:

α : là learning rate, biểu thị tốc độ cập nhật background, được chọn cố định.

- α lớn: đặc trưng cho sự cập nhật nhanh (quick update), thích hợp cho phát hiện chuyển động nhanh. ($\alpha > 0.02$).
- α nhỏ : đặc trưng cho sự ổn định (stability), thích hợp cho phát hiện chuyển động chậm ($\alpha < 0.02$).

d : là khoảng cách Euclidean giữa giá trị trung bình của phân phối và giá trị pixel hiện tại.

- **Phân loại background/foreground:**

$$\frac{|x_t - \mu_t|}{\sigma_t} > k \rightarrow \text{foreground pixel}$$

$$\frac{|x_t - \mu_t|}{\sigma_t} \leq k \rightarrow \text{background pixel} \quad (3.4)$$

Trong đó k là ngưỡng chỉ mức độ dao động cho phép của giá trị pixel background so với giá trị trung bình của phân phối (k thường chọn là 2.5)

- k lớn : thì cho phép giá trị mức xám của pixel biến động mạnh, đối tượng chuyển động cần có mức xám chênh lệch lớn so với nền \rightarrow loại được nhiều nhiễu, thường áp dụng cho khung cảnh video rõ nét.
- k nhỏ: biểu thị nền ít biến động, phát hiện tốt các
- biến động nhỏ, thích hợp cho phát hiện đối tượng nhỏ, mờ, tuy nhiên gây ra rất nhiều nhiễu.

Khi một pixel là foreground thì giá trị của nó không tuân theo phân phối, vì vậy không nên sử dụng giá trị này để cập nhật mô hình Gaussian. Ta chỉ cập nhật giá trị trung bình của phân phối đối với pixel được phân loại là background.

$$\mu_t = M \mu_{t-1} + (1-M)(\alpha x_t + (1-\alpha)\mu_{t-1}) \quad (3.5)$$

Với $M = 1$ nếu pixel là foreground, $M = 0$ nếu pixel là background.

- ❖ **Module hậu xử lý (Post-Processing)**

Thuật toán hậu xử lý phân tích ảnh foreground, khoanh vùng đưa ra tọa độ các đối tượng chuyển động.

Do tác động của nhiễu, ảnh nền thường không ổn định hoàn toàn mà có sự dời chỗ nhỏ hoặc có các chuyển động nhỏ không mong muốn, khi đó một số vấn đề sau sẽ nảy sinh:

- Nhiều foreground pixel không thuộc đối tượng chuyển động thực tế
- Đối tượng chuyển động bị chia mảnh do xuất hiện các vùng đen (background pixel) giữa vùng trắng (đối tượng chuyển động)
- Đối tượng kích thước lớn bị chia mảnh làm nhiều vùng trắng riêng biệt
- Xuất hiện các vùng trắng do các chuyển động nhỏ, tạm thời gây ra.

Thuật toán hậu xử lý sẽ lọc các vùng trắng trong ảnh, giảm xác suất phát hiện nhầm.

- **Pixel Level Processing**

Mục đích : loại bỏ các foreground pixel sinh ra bởi nhiễu

Sử dụng các phép biến đổi hình thái học (morphological) : phép ăn mòn (erosion) sau đó đến phép giãn nở (dilation).

- Sử dụng phép ăn mòn: loại bỏ các vùng trắng nhỏ cô lập
- Sử dụng phép giãn nở: loại bỏ các hồ đen trong vùng trắng, bù phần đường bao của đối tượng bị ăn mòn.

- **Connected Region**

Ở các bước trên, ta chỉ làm việc với ảnh foreground, ở bước này các vùng đối tượng sẽ được xác định. Mỗi vùng đối tượng đại diện bởi tập các điểm trên đường bao xấp xỉ của vùng. Các đối tượng chuyển động lớn có thể bị chia thành các vùng nhỏ. Phương pháp đơn giản nhất là nối các vùng trắng gần nhau lại. Sau đó một số phép xử lý liên quan đến các vùng đối tượng được áp dụng gồm:

- Kiểm tra số lượng đối tượng hiện có trong frame ảnh : nếu số lượng đối tượng vượt quá ngưỡng thì bỏ qua phát hiện chuyển động tại frame này.
- Kiểm tra kích thước (diện tích, chu vi) của vùng đối tượng : nếu lớn vượt qua một ngưỡng thì vùng này không được xem là chuyển động.

- **Region Processing**

Đường bao đối tượng chứa rất nhiều điểm, thực sự không cần thiết cho xử lý phát hiện, ta chỉ cần lấy hình chữ nhật bao quanh đối tượng. Ở bước này mỗi đối tượng sẽ coi như ứng với một hình chữ nhật trên frame ảnh. Tuy nhiên, nếu coi các hình chữ nhật là các đối tượng thực sự thì sẽ không loại được các nhiễu do chuyển động nhỏ gây ra . Ở bước này, các hình chữ nhật sẽ được gán định danh, biểu thị nó thuộc về đối tượng nào ở các frame trước đó.

- Nếu hình chữ nhật này là vị trí tiếp theo của một đối tượng, thì cập nhật vị trí mới này cho đối tượng
- Nếu hình chữ nhật này là cô lập, không gần một đối tượng đã gán định danh nào trước đó, thì một định danh mới sẽ được tạo và gán cho hình chữ nhật.

Đi kèm đó là thuật toán dự đoán quỹ đạo, được dùng khi một đối tượng bị mất dấu, vị trí dự đoán sẽ được tính và cập nhật vào quỹ đạo, giúp duy trì theo dõi được đối tượng khi đối tượng có trở lại.

- ❖ **Đánh giá thuật toán**

Việc đánh giá một thuật toán bất kỳ thường dựa trên 3 yếu tố: độ phức tạp tính toán; bộ nhớ sử dụng và hiệu quả (chất lượng) của thuật toán.

- Độ phức tạp tính toán: $O(n)$ với n là kích thước frame ảnh (số lượng pixel được duyệt).
- Bộ nhớ trong lưu trữ như bảng dưới đây:

Dữ liệu	Dung lượng
---------	------------

Input frame buffer	720x576x8 = 3.1MB
Output frame buffer	720x576x8 = 3.1MB
Variables	10*32bit
Internal memory	4MB cache

- Hiệu quả tính toán (xem bảng 3.1)

2. Triển khai thuật toán

Thuật toán phát hiện mục tiêu chuyển động (RGA) đã được viết trên ngôn ngữ C++ và triển khai trên nền tảng DSP của thiết bị Apollo17.

Kết quả đánh giá cụ thể như sau:

Bảng 3. 1: Kết quả đánh giá thuật toán phát hiện chuyển động

STT	Chỉ tiêu, tính năng	Thông số	Đơn vị
1	Xác suất phát hiện đối tượng trong điều kiện thường	94,2	%
2	Xác suất phát hiện đối tượng trong điều kiện ánh sáng yếu < 15 lux	92,8	%
3	Xác suất phát hiện đối tượng với camera nhiệt	96,0	%
4	Kích thước đối tượng nhỏ nhất có thể phát hiện	8x10	Pixel
5	Xác suất phát hiện nhầm đối tượng	4,5	%
6	Số lượng đối tượng chuyển động có thể phát hiện đồng thời	30	Đối tượng
7	Độ nhạy thuật toán theo mức xám	20/255	Mức
8	Tốc độ xử lý trung bình	100	Hình/giây

3. Một số hình ảnh thử nghiệm



Hình 3. 4: Phát hiện trong điều kiện thiếu sáng



Hình 3. 5: Phát hiện với ảnh nhiệt

KẾT LUẬN

🚦 Những kết quả đạt được

Đề tài đã thực hiện thành công nhiệm vụ đặt ra với mục tiêu chính:

+ *Nghiên cứu, thiết kế thành công bo mạch phân cứng Apollo17 (dạng KIT phát triển) cho phép thực thi các thuật toán xử lý ảnh phức tạp thay cho máy tính.*

+ Thiết kế đã được chế tạo thử nghiệm, chứng minh hoạt động, đáp ứng 100% chỉ tiêu yêu cầu.

🚦 Hướng phát triển tiếp theo

Hướng phát triển tiếp theo của đề tài này đó là:

- Tối ưu kích thước bo mạch xuống dưới 10x10 cm. Trang bị cho các máy tính nhúng phục vụ mục đích xử lý hình ảnh trong các ứng dụng Quân sự và Dân sự.



Hình 4. 1: Minh họa máy tính nhúng dùng trong Quân sự

- Trang bị các đầu vào tốc độ cao (>100fps) phục vụ mục đích lắp đặt trên các máy bay không người lái UAV. Cho phép xử lý hình ảnh trong quá trình bay trinh sát.



Hình 4. 2: Minh họa ứng dụng cho máy bay UAV

- Các kết quả nghiên cứu của đề tài này có thể sử dụng phục vụ mục đích đào tạo, phát triển công nghệ xử lý ảnh trên nền tảng nhúng.

TÀI LIỆU THAM KHẢO

Tiếng Việt

- [1]. TS. Trần Quang Vinh, ThS.Chữ Văn An, “*Nguyên lý kỹ thuật Điện tử*”. NXB Giáo dục 2005.
- [2]. Phạm Văn Khiên, Phạm Trần Hương Giang, “*Báo cáo đánh giá thuật toán phát hiện đối tượng chuyển động*”, Viettel R&D 2016.

Tiếng Anh

- [3]. Tammy Noergaard, “*Embedded Systems Architecture*”, Elsevier 2005.
- [4]. Shehrzad Qureshi, “*Embedded Image Processing on the TMS320C6000TM DSP*”, Springer Science+Business Media, Inc. 2005.
- [5]. Branislav Kisacanin, Shuvra S.Bhattacharyya, Sek Chai, “*Embedded Computer Vision*”, Springer 2009.
- [6]. Christopher Hallinan, “*Embedded Linux Primer*”, Pearson Education, Inc. 2011.
- [7]. Augusto Vega, Pradip Bose, Alper Buyuktosunoglu, “*Rugged Embedded Systems: Computing in Harsh Environment*”, Elsevier 2017.

Internet

- [8]. Trang wiki của hãng Texas Instrument
http://processors.wiki.ti.com/index.php/DM816x_AM389x_PSP_User_Guide
- [9]. TMS320DM8168/Technical Documents
<http://www.ti.com/product/TMS320DM8168/technicaldocuments>
- [10]. TMS320DM8168/Tool&Software
<http://www.ti.com/product/TMS320DM8168/toolssoftware>
- [11]. DDR3 point-to-point design support
https://www.micron.com/~/media/documents/products/technical-note/dram/tn4113_ddr3_point_to_point_design.pdf
- [12]. Bộ công cụ hỗ trợ phát triển phần mềm Linux EZSDK
<http://www.ti.com/tool/linuxezsdk-davinci>

Sản phẩm tham khảo

- (1) Video Tracker 8200: http://www.eoimaging.com/documents/8200_R1.pdf
- (2) IGEPv2: <https://www.isee.biz/products/igep-processor-boards/igepv2-dm3730>
- (3) OZ745: <https://www.avnet.com/shop/us/p/kits-and-tools/development-kits/avnet-engineering-services/aes-oz745-3074457345623375756/>
- (4) Z3-DM8169-VI-RPS: <http://z3technology.com/oem/Product/Z3-DM8169-VI-RPS/32.html>

PHỤ LỤC: HỒ SƠ THIẾT KẾ NGUYÊN LÝ SẢN PHẨM