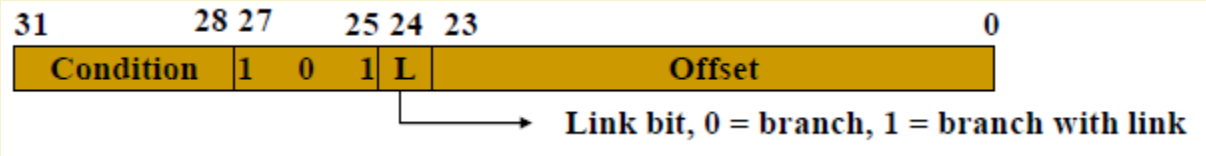
Branch Instruction

- All branches are relative to the program counter
- Jump is always within a limit of ±32MB
- Conditional branches are formed by using condition codes
- Subroutine call is also modeled as a variant of branch instruction
- Two opcodes
  - B – standard branch
  - BL – branch with link, PC+4 saved in link register R14
  - BL can be used for subroutine call
  - Return from subroutine can be affected by copying R14 into R15

NPTEL

Branch Instruction Format



Offset calculation:

- Compute 26-bit difference between branch instruction and target
- Right shift by 2-bits, as instruction are always word aligned, least significant two bits are always zeros
- 24-bit value stored with the instruction
- During execution, offset left shifted by 2 bits, sign extended to 32 bits, added to PC to get the branch target

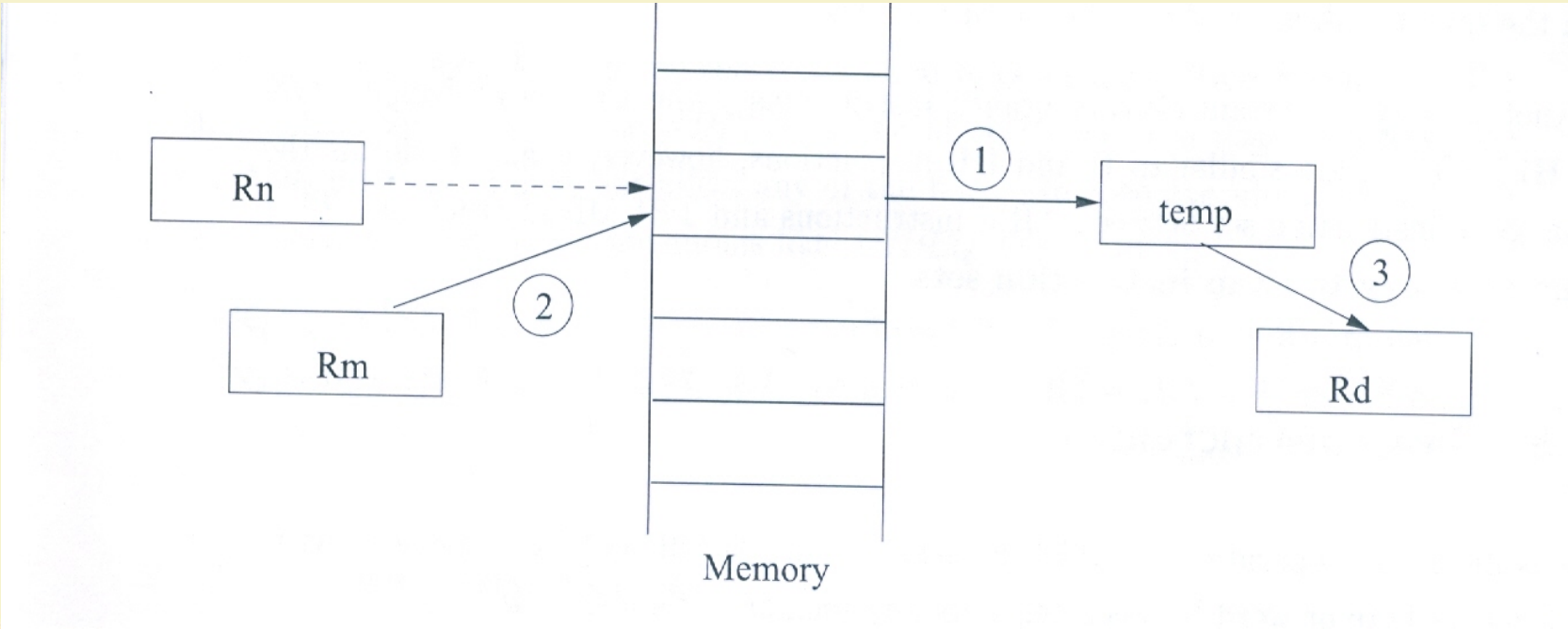
NPTEL

Swap Instruction

- Atomic operation in which a memory read is followed by a memory write
- Moves byte or word between registers and memory
- Format:
  - SWP Rd, Rm, [Rn]
  - SWPB Rd, Rm, [Rn]
- Two cycle operation
- Content of memory location pointed to by register Rn is copied into a temporary space
- Content of register Rm is copied into the memory location
- Content of the temporary space is copied into the register Rd
- To effect an interchange between Rd and Rm, they should be made the same register

NPTEL

Swap Instruction Execution



Modifying Status Registers

- Status registers can only be modified indirectly
- MSR moves content from CPSR/SPSR to selected general purpose register
- MRS copies the content of selected general purpose register to CPSR/SPSR
- Can only be executed in privileged mode

NPTEL

THUMB Instructions

- THUMB instructions are 16-bit long
- Stored in a compressed form
- Executed unconditionally, excepting the branch instructions
- Have unlimited access to registers R0-R7 and R13-R15
- A reduced number of instructions can access the full register set
- No MSR/MRS instructions
- Maximum number of SWI calls limited to 256
- Instructions look more like a conventional processor's instructions. For example, PUSH and POP instructions
- On reset or exception, processor enters into ARM instruction set mode
- To switch between instruction sets BX / BLX has to be used

NPTEL

ARM vs. THUMB

Similarities:

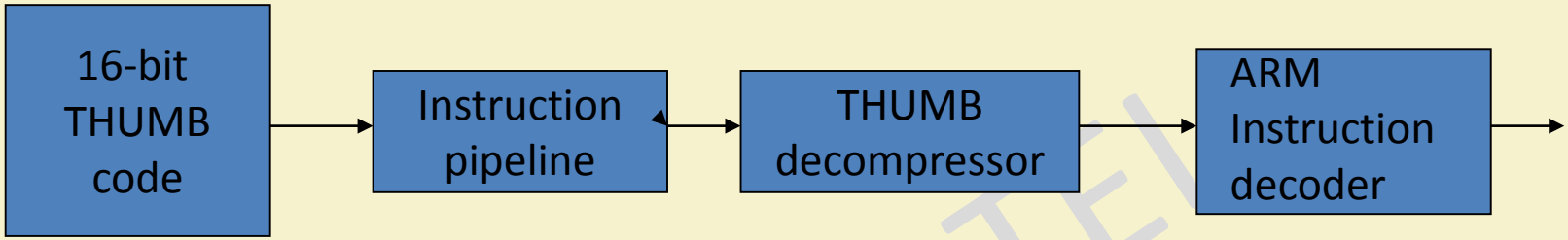
- Load-Store architecture
- Support for 8-, 16-, and 32-bit data types
- 32-bit unsegmented memory

Differences:

- Conditional execution
- ARM 3-address format, THUMB 2-address format
- THUMB has less regular instruction format
- THUMB has explicit shift opcodes, in ARM only operand modifiers

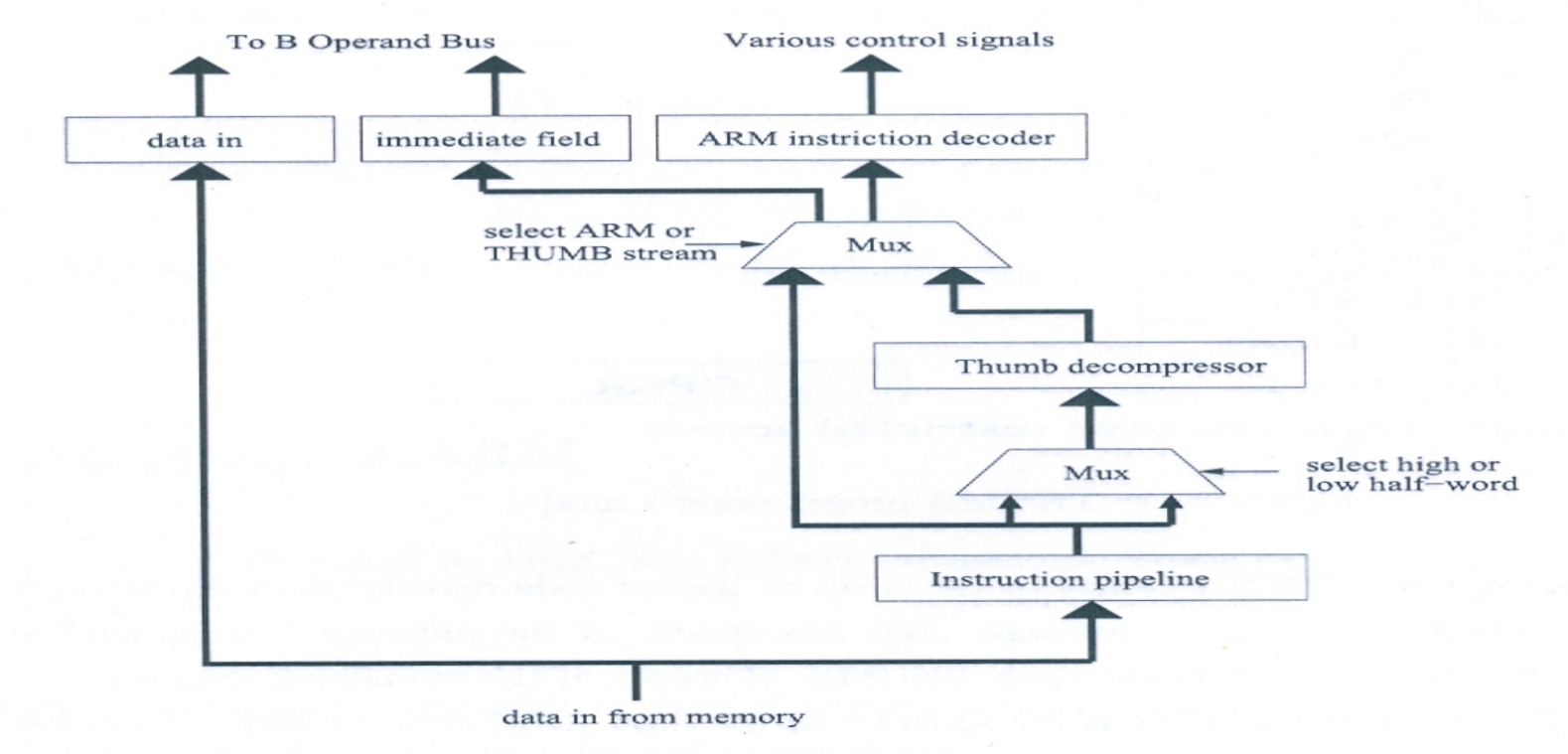
NPTEL

THUMB Instruction Processing



NPTEL

ARM decompressor



Advantages of THUMB

- Higher code density
- THUMB code, on an average, requires 30% less space
- If memory is organized as 32-bit words, ARM code is 40% faster than THUMB
- If memory is organized as 16-bit words, THUMB code is 45% faster than ARM
- THUMB code uses upto 30% less power than ARM code

NPTEL

- For best performance, use 32-bit memory and ARM instruction set
- For best cost and power efficiency, use 16-bit memory with THUMB code
- In a typical embedded system,
  - ARM be used in 32-bit on-chip memory for small speed-critical routines
  - THUMB code should be used in 16-bit off-chip memory for large non-critical control routines

NPTEL

Exceptions in ARM

- Can occur from various situations
  - Through execution of instructions – software interrupts, undefined instructions, memory abort
  - Side-effect of instruction – data fetch failure
  - Other sources like RESET, FIQ, IRQ interrupts
- Processor switches to privileged mode
- Current value of PC+4 saved in the link register
- Current CPSR copied into SPSR of the privileged mode
- IRQ interrupts are disabled
- If exception has been caused by FIQ, FIQ interrupts are also disabled
- Program counter loaded with exception vector address
- Execution of exception starts

NPTEL

ARM7 Vector Table

Exception type	Mode	Vector address
Reset	Supervisor	0x00000000
Undefined instruction	Undefined	0x00000004
Software interrupt (SWI)	Supervisor	0x00000008
Prefetch abort (instruction fetch abort)	Abort	0x0000000C
Data abort (data access memory abort)	Abort	0x00000010
IRQ (interrupt)	IRQ	0x00000018
FIQ (fast interrupt)	FIQ	0x0000001C

NPTEL

Issues Related to Exceptions

- Vector at location 0x0000014 is missing to ensure a backward compatibility
- FIQ has been given the highest address
  - Interrupt service routine can start from this address directly
  - Eliminates the necessity of another jump instruction from the vector address to the ISR, thus saving time needed to start the routine after the fast interrupt has occurred

NPTEL

Finiding the maximum of a set of numbers

```
        EOR R1, R1, R1          ;clear R1 to store the largest
        CMP R2, #0
        BEQ Over                ;if block is empty, done
Loop
        LDR R3, [R0]  ;get the data
        CMP R3, R1    ;do comparison
        BCC Looptest ;skip if R1 is bigger
        MOV R1, R3    ;else get the larger in R1
Looptest
        ADD R0, R0, #4          ;increment pointer R0
        SUBS R2, R2, #1         ;decrement number of elements left
        BNE Loop                ;if not done, loop
Over
                                ;R1 holds the largest
```

NPTEL

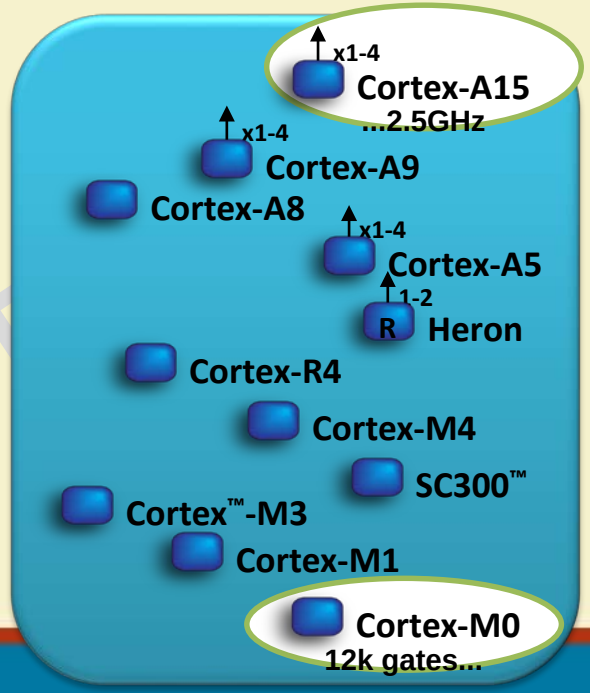
Comparing two null terminated strings

```
Loop
        LDRB R3, [R0]     ;get next character of string 1
        LDRB R4, [R1]     ;get next character of string 2
        CMP R3, R4        ;compare
        BNE Notsame       ;if not same, strings do not match
        CMP R3, #0        ;check if end of string reached
        BEQ Same          ;if equal, same
        ADD R0, R0, #1    ;increment pointer to string 1
        ADD R1, R1, #1     ;increment pointer to string 2
        BAL Loop          ;branch always to check next character
Notsame
        MOV R2, #-1       ;mark not matched
        BAL Over
Same
        MOV R2, #0        ;mark matched
Over
                                        ;R2 holds the match
```

NPTEL

ARM Cortex Processors (v7)

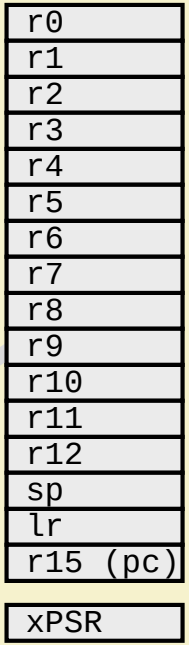
- ARM Cortex-**A** family (v7-A):
  - Applications processors for full OS and 3rd party applications
- ARM Cortex-**R** family (v7-R):
  - Embedded processors for real-time signal processing, control applications
- ARM Cortex-**M** family (v7-M):
  - Microcontroller-oriented processors for MCU and SoC applications



NPTEL

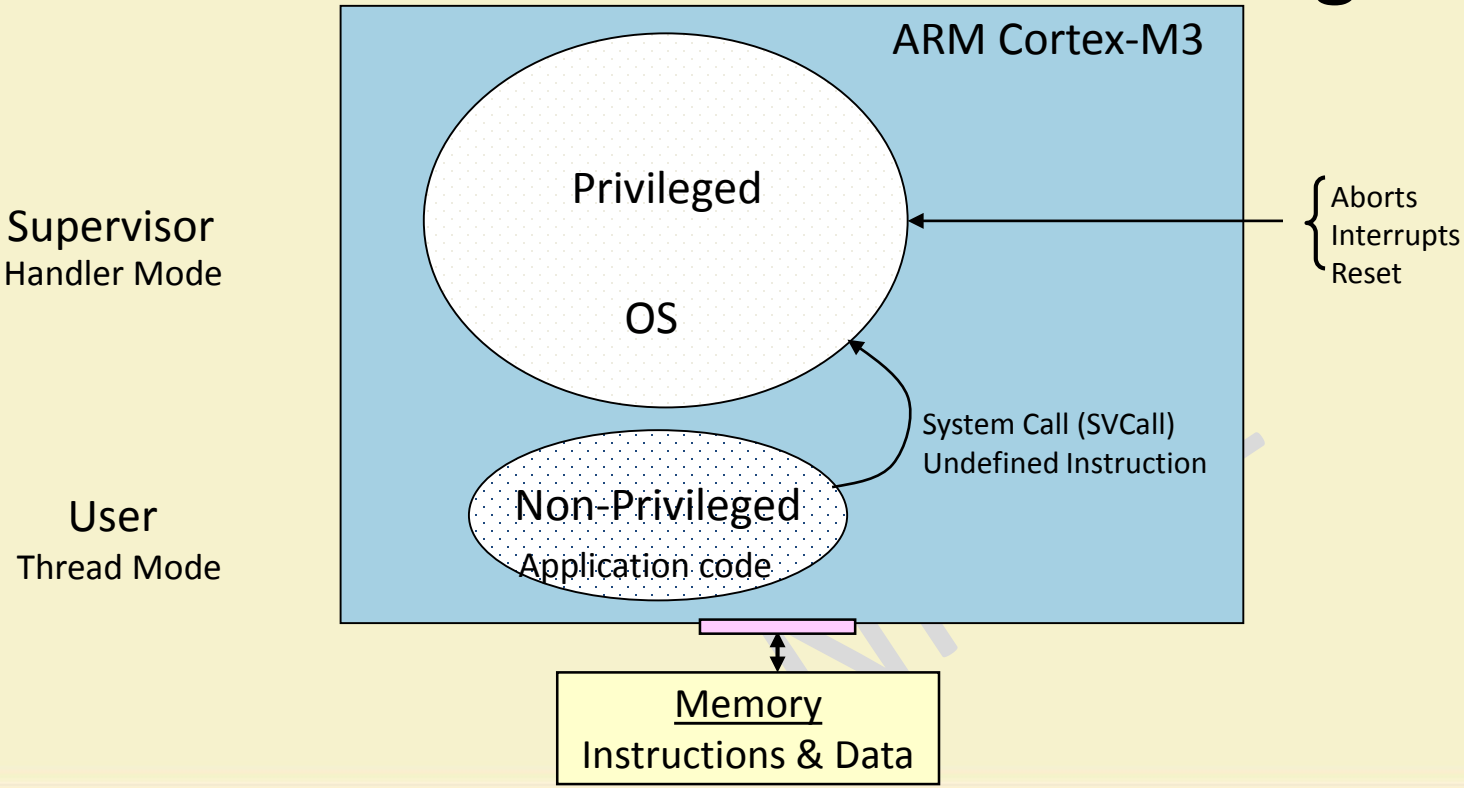
Cortex-M Programmer's Model

- Fully programmable in C
- Stack-based exception model
- Only two processor modes
  - Thread Mode for User tasks
  - Handler Mode for OS tasks and exceptions
- Vector table contains addresses



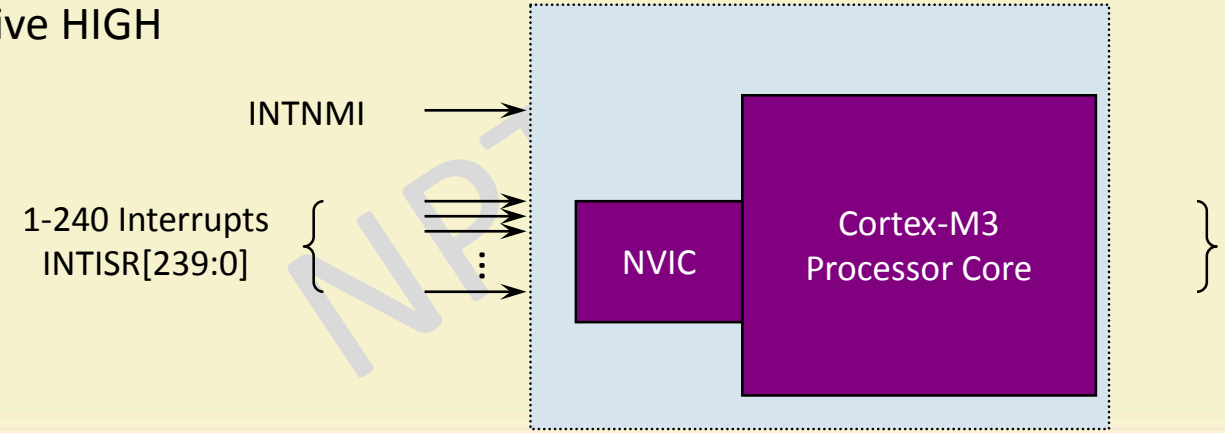
NPTEL

Cortex-M3 Processor Privilege



Cortex-M3 Interrupt Handling

- One Non-Maskable Interrupt (INTNMI) supported
- 1-240 prioritizable interrupts supported
  - Interrupts can be masked
  - Implementation option selects number of interrupts supported
- Nested Vectored Interrupt Controller (NVIC) is tightly coupled with processor core
- Interrupt inputs are active HIGH



IIT KHARAGPUR
NPTEL ONLINE
CERTIFICATION COURSES

Cortex-M3 Exception Handling

- **Reset :** power-on or system reset
- **NMI :** cannot be stopped or preempted by any exception other than reset
- **Faults**
  - **Hard Fault** : default Fault or any fault unable to activate
  - **Memory Manage** : MPU violations
  - **Bus Fault** : prefetch and memory access violations
  - **Usage Fault** : undef instructions, divide by zero, etc.
- **SVCall :** privileged OS requests
- **Debug Monitor :** debug monitor program
- **PendSV :** pending SVCalls
- **SysTick Interrupt :** internal sys timer, i.e., used by RTOS to periodically check resources or peripherals
- **External Interrupt :** i.e., external peripherals

NPTEL

Conditional Execution

- If – Then (IT) instruction added (16 bit)
  - Up to 3 additional “then” or “else” conditions maybe specified (T or E)
  - Makes up to 4 following instructions conditional

```
ITTET EQ
Inst 1
Inst 2
Inst 3
Inst 4
```

```
MOVEQ
ADDEQ
SUBNE
ORREQ
```

- Any normal ARM condition code can be used
- 16-bit instructions in block do not affect condition code flags
  - Apart from comparison instruction
  - 32 bit instructions may affect flags (normal rules apply)
- Current “if-then status” stored in CPSR
  - Conditional block maybe safely interrupted and returned to
  - Must NOT branch into or out of ‘if-then’ block

NPTEL

PIC Microcontroller – PIC18F Family

Santanu Chattopadhyay
Electronics and Electrical Communication Engineering

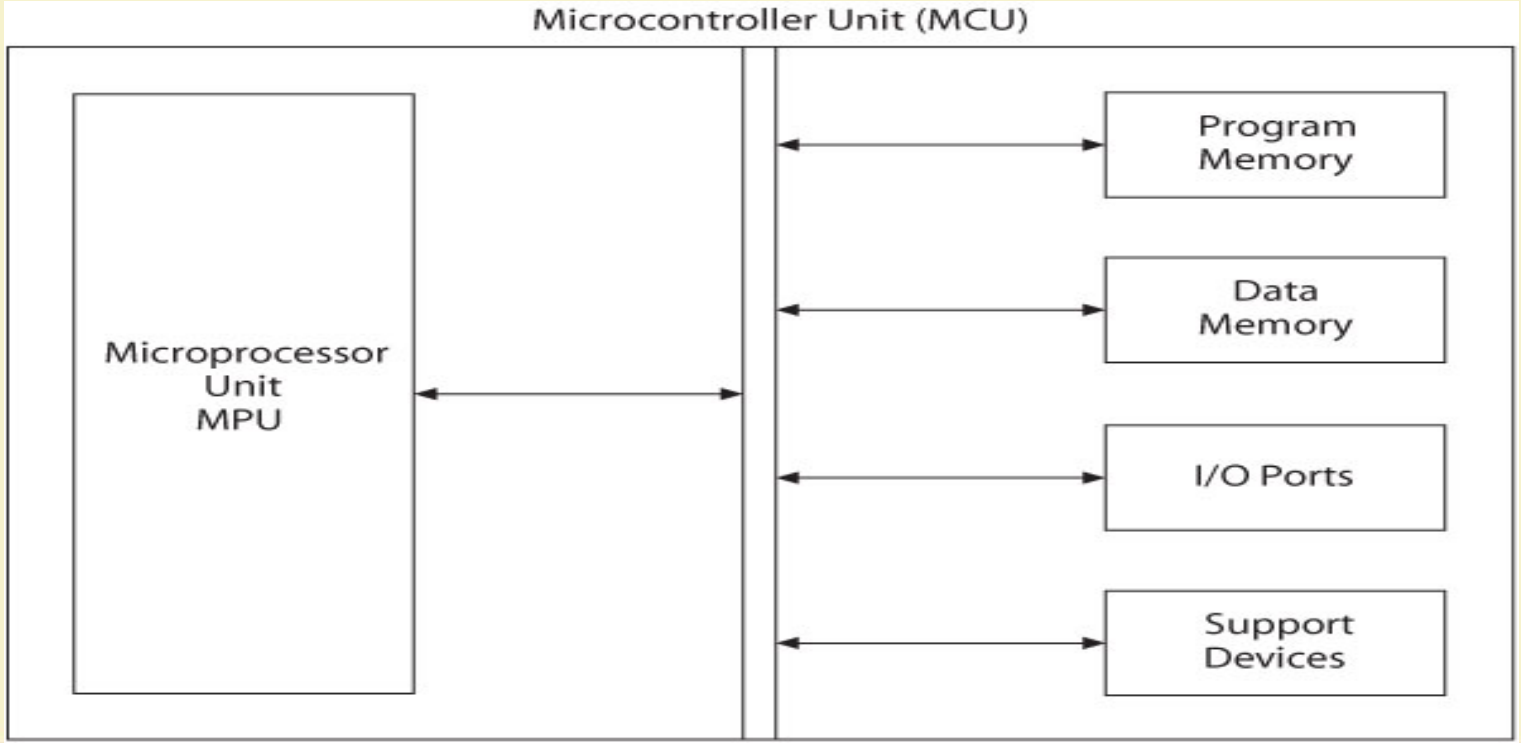
NPTEL

PIC18F Microcontrollers

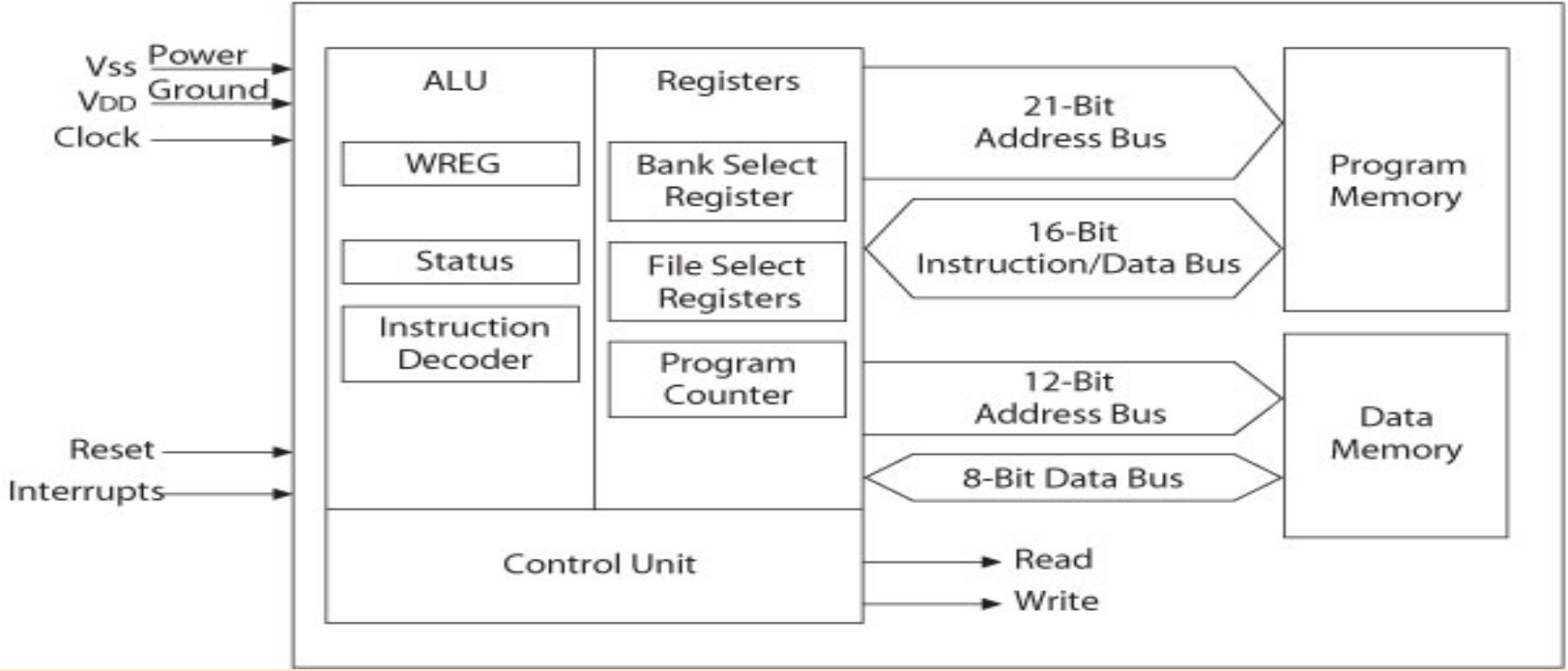
- Microcontroller Unit (MCU)
  - Microprocessor unit (MPU)
  - Harvard Architecture
    - Program memory for instructions
    - Data memory for data
  - I/O ports
  - Support devices such as timers

NPTEL

Microcontroller Unit



PIC18F – MPU and Memory



Microprocessor Unit

- Includes Arithmetic Logic Unit (ALU), Registers, and Control Unit
  - Arithmetic Logic Unit (ALU)
    - Instruction decoder
      - 16-bit instructions
    - Status register that stores flags
      - 5-bits
    - WREG – working register
      - 8-bit accumulator

NPTEL

Microprocessor Unit

- Registers
  - Program Counter (PC)
    - 21-bit register that holds the Program Memory address
  - Bank Select Register (BSR)
    - 4-bit register used in direct addressing the Data Memory
  - File Select Registers (FSRs)
    - 12-bit registers used as memory pointers in indirect addressing Data Memory
- Control unit
  - Provides timing and control signals
    - Read and Write operations

NPTEL

PIC18F - Address Buses

- Address bus
  - 21-bit address bus for Program Memory
    - Addressing capacity: 2 MB
  - 12-bit address bus for Data Memory
    - Addressing capacity: 4 KB

NPTEL

Data Bus and Control Signals

- Data bus
  - 16-bit instruction/data bus for Program Memory
  - 8-bit data bus for Data Memory
- Control signals
  - Read and Write

NPTEL

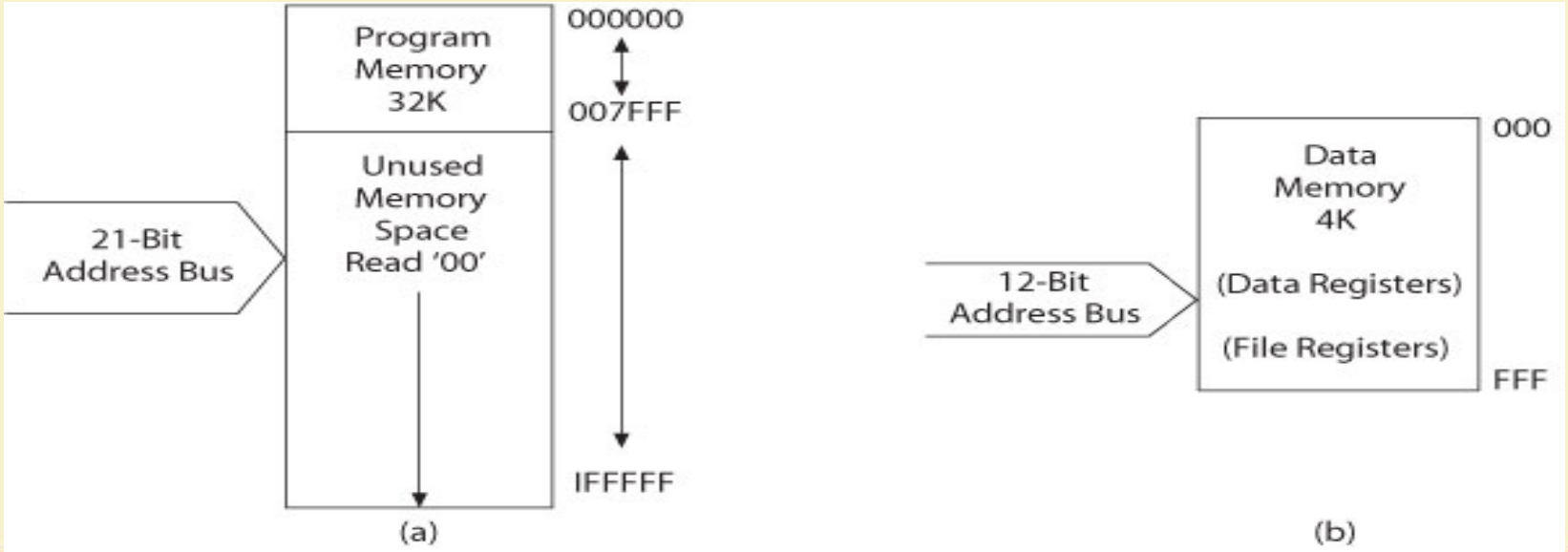
PIC18F452/4520 Memory

- Program Memory: 32 K
  - Address range: 000000 to $007FFF_H$
- Data Memory: 4 K
  - Address range: 000 to FFF_H
- Data EEPROM
  - Not part of the data memory space
  - Addressed through special function registers

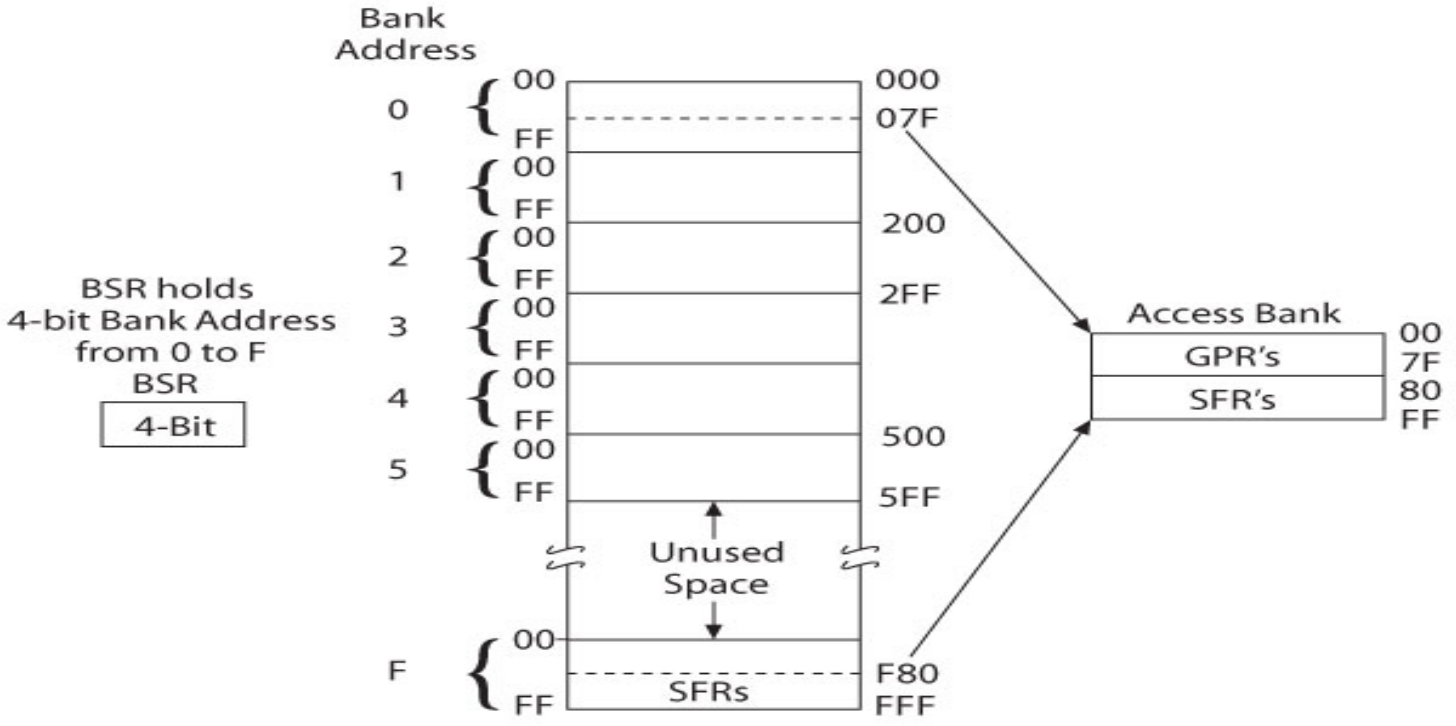
NPTEL

PIC18F452/4520 Memory

- Program Memory
- Data Memory



Data Memory Banks

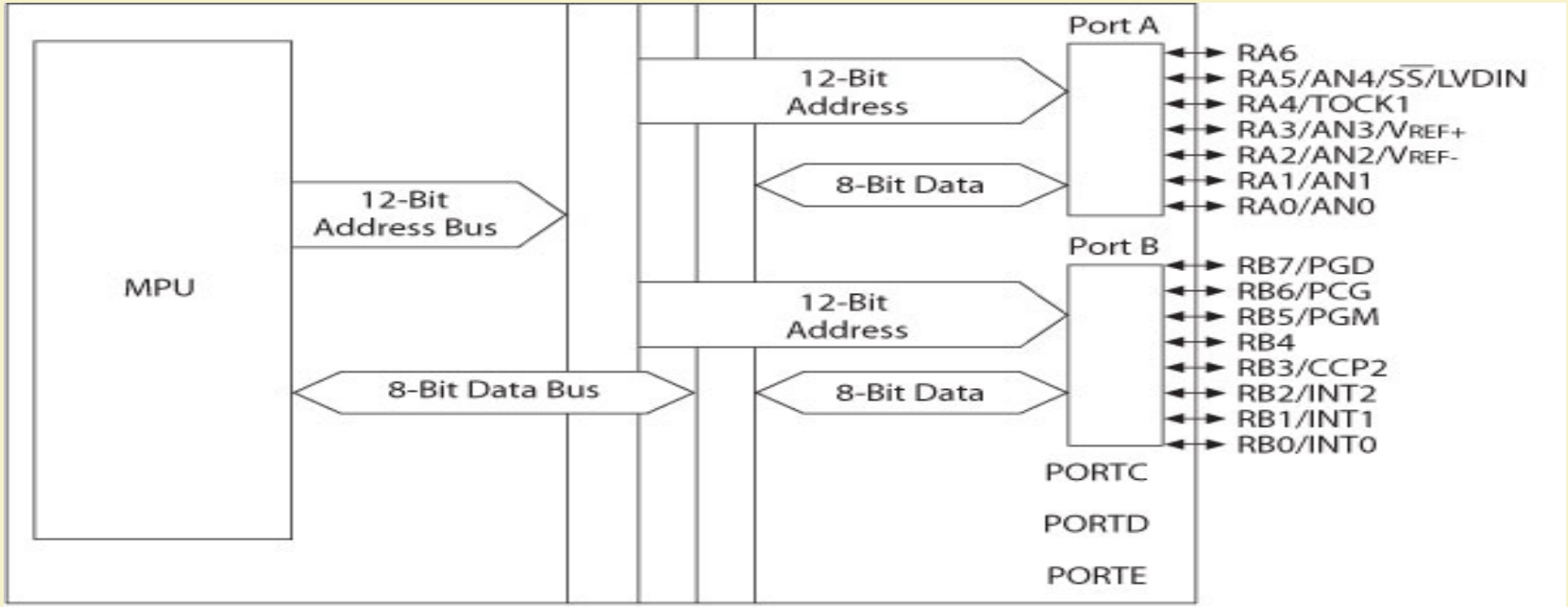


PIC18F452 I/O Ports

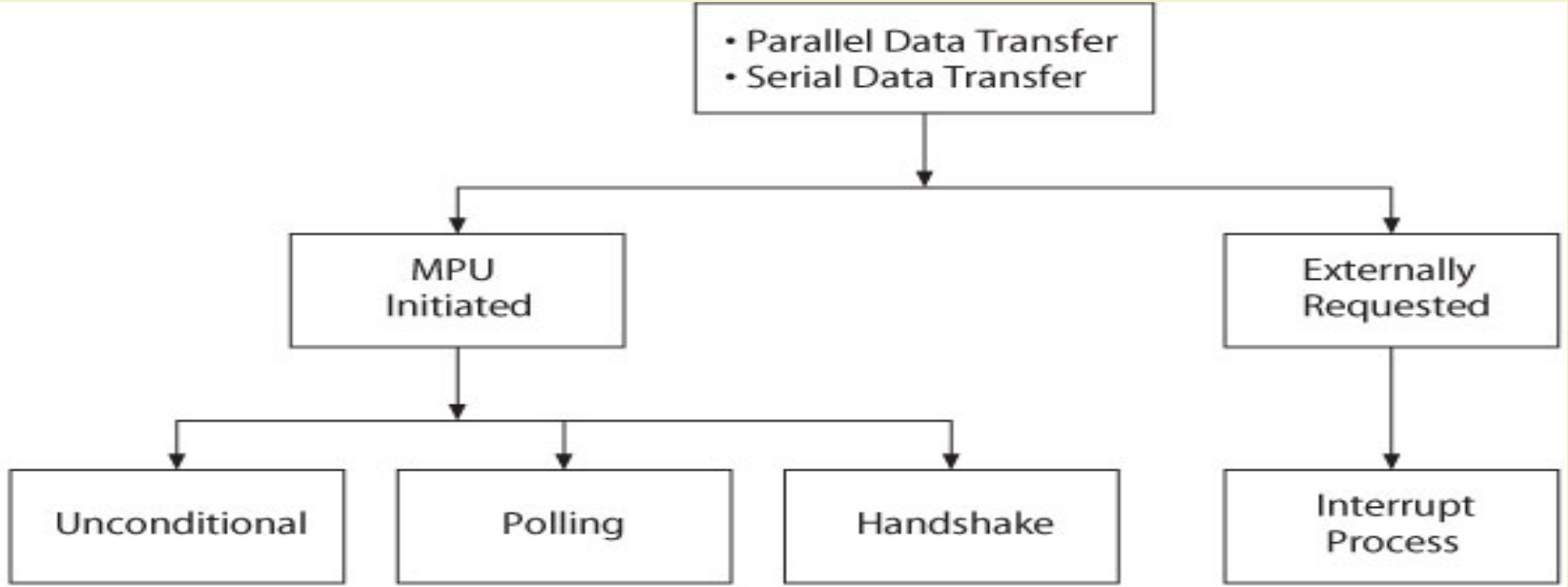
- Five I/O ports
  - PORT A through PORT E
  - Most I/O pins are multiplexed
  - Generally have eight I/O pins
  - Addresses already assigned to these ports
  - Each port is identified by its assigned SFR

NPTEL

I/O Ports A and B



Data Transfer

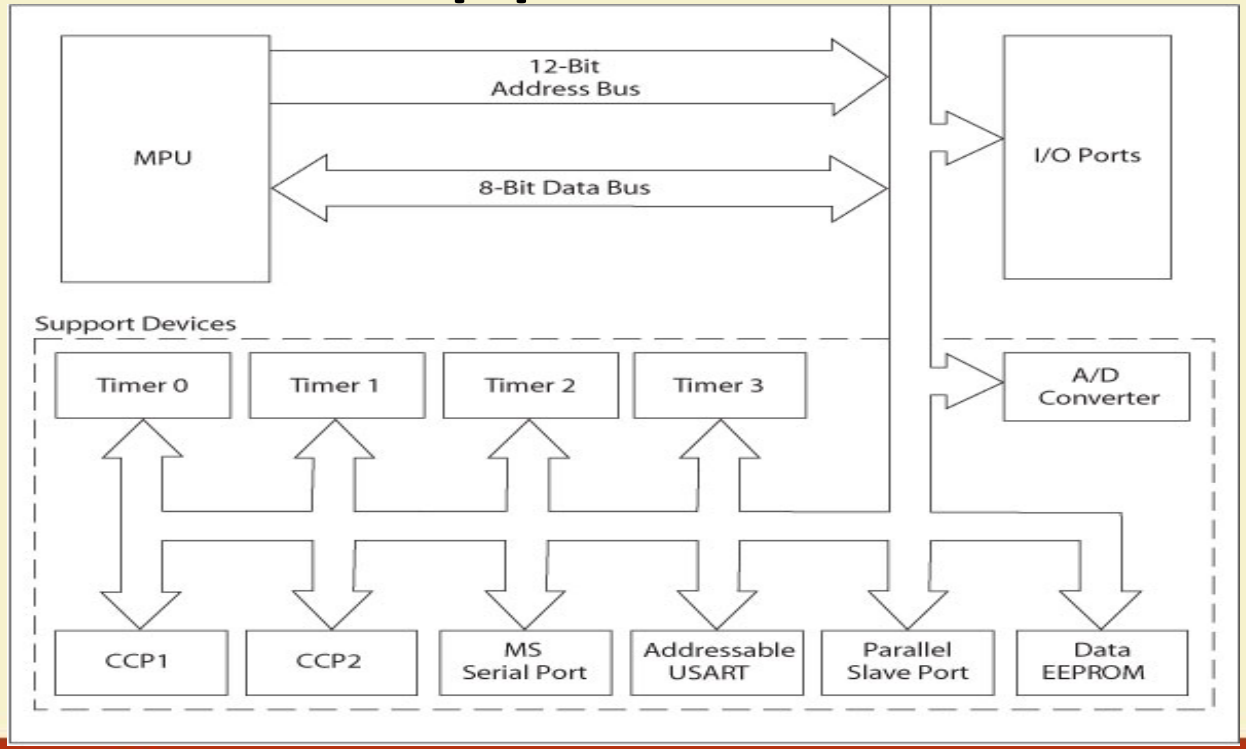


MCU Support Devices

- Timers
  - Capture, Compare and PWM (CCP Modules)
- Serial Communications
  - Master Synchronous Serial Port (MSSP)
  - Addressable USART
- A/D converter
- Parallel Slave Port (PSP)
- Data EEPROM

NPTEL

MCU Support Devices

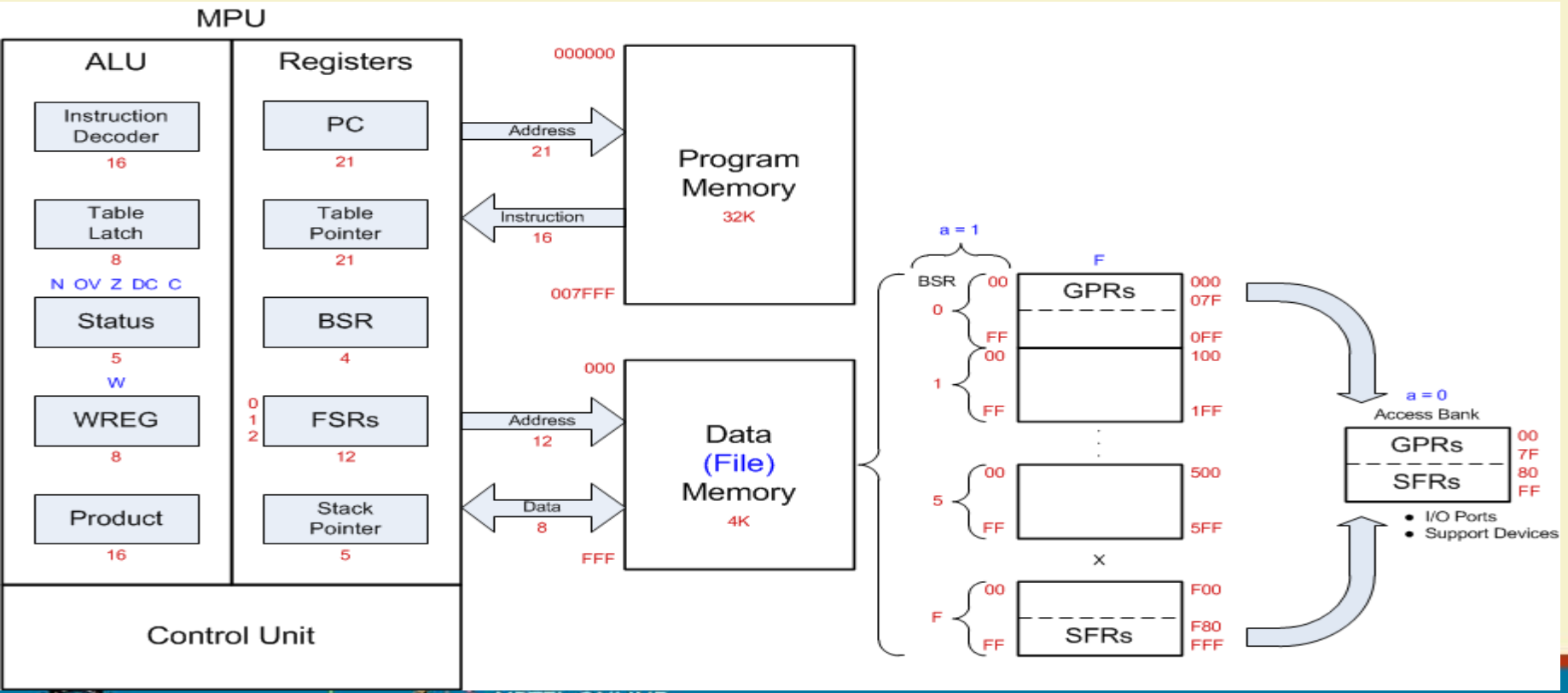


PIC18F Special Features

- Sleep mode
- Watchdog timer (WDT)
- Code protection
- In-circuit serial programming
- In-circuit debugger

NPTEL

PIC18F452 Programming Model



List of Selected Microcontroller Families from Microchip

Part No.	Program OTP/Flash	EE PROM	RAM	Total Pins	I/O Pins	Analog ADC	Analog Comp.	Digital Timers/ WDT	Digital Serial I/O	CCP/ ECCP	Max Speed MHz	Instruc-tion Size	Total Instruc-tions
10F200	256x12 Flash		16	8	4			1-8 bit, 1-WDT			4	12-bit	33
10F220	256x12 Flash		16	8	4	2x8-bit		1-8 bit, 1-WDT			8	12-bit	33
12F510	1536x12 Flash		38	8	6	3x8-bit	1	1-8 bit 1-WDT			8	12-bit	33
16F506	1536x12 Flash		67	14	12	3x8-bit	2	1-8 bit 1-WDT			20	12-bit	33
16C55A	768x12 OTP		24	28	20			1-8 bit 1-WDT			40	12-bit	33
16CR58B	3072x12 ROM		73	18	12			1-8 bit 1-WDT			20	12-bit	33
12F683	2048x14 Flash	256	128	8	6	4x10-bit	1	1-16 bit, 2-8 bit, 1-WDT			20	14-bit	35
16F687	2048x14 Flash	256	128	20	18	12x10-bit	2	1-16 bit, 1-8 bit, 1-WDT	EU/I^{2}C/ SPI		20	14-bit	35
18F1230	2048x16 Enh Flash	128	256	18-28	16	4x10-bit	3	2-16 bit 1-WDT	EU		40	16-bit	77
18F4520	16384x16 Enh Flash	256	1536	40-44	36	13x10-bit	2	1-8 bit, 3-16 bit, 1-WDT	EU/ MI^{2}C /SPI	1/1	40	16-bit	77
18F6527	24576x16 Enh Flash	1024	3936	64	54	12x10-bit	2	2-8 bit, 3-16 bit, 1-WDT	2EU/ 2 - MI^{2}C /SPI	2/3	40	16-bit	77
18F8622	32768x16 Enh Flash	1024	3936	80	70	16x10-bit	2	2-8 bit, 3-16 bit, 1-WDT	2EU/ 2 - MI^{2}C /SPI	2/3	40	16-bit	77
18F96J60	32768x16 Flash		2048	100	72	16x10-bit	2	2-8 bit, 3-16 bit, 1-WDT	2EU/ 2 - MI^{2}C /SPI	2/3	42	16-bit	77
24FJ128GA-010	65536x16 Flash		8192	100-128	86	16x10-bit	2	5-16 bit, 1-WDT	2 –UART 2-I^{2}C/ SPI	5	32	16-bit	77

Abbreviations: 1) ADC: Analog-Digital Converter, 2) AUSART: Addressable USART, 3) CCP: Capture/Compare/PWM, 4) ECCP: Enhanced CCP, 5) EU: Enhanced USART, 6))Enh Flash: Enhanced Flash, 7) I^{2}C: Inter-integrated Circuit Bus, 8) MI^{2}C/SPI: Master I^{2}C /SPI, 9) OTP: One-Time Programmable, 10) SPI: Serial Peripheral Interface, 11) USART: Universal Synchronous/Asynchronous Receiver/Transmitter, 11) WDT: Watchdog Timer

PIC18F Instructions

- 77 assembly language instructions
  - Earlier PIC families have 33 or 35 instructions
- PIC18F instruction set
  - Most instructions are 16-bit word length
  - Four instructions are 32-bit length

NPTEL

Instruction Descriptions

- Copy (Move) 8-bit number (Literal) into W register
  - Mnemonics: MOVLW 8-bit
  - Binary format:

  0000 1110 XXXX XXXX (any 8-bit number)
- Copy (Move) contents of W register into PORTC (File)
  - Mnemonics: MOVWF PORTC, a
    - ('a' indicates that PORTC is in the Access Bank)
  - Binary format:

  0110 1110 1000 0010 (82_H is PORTC address)

NPTEL

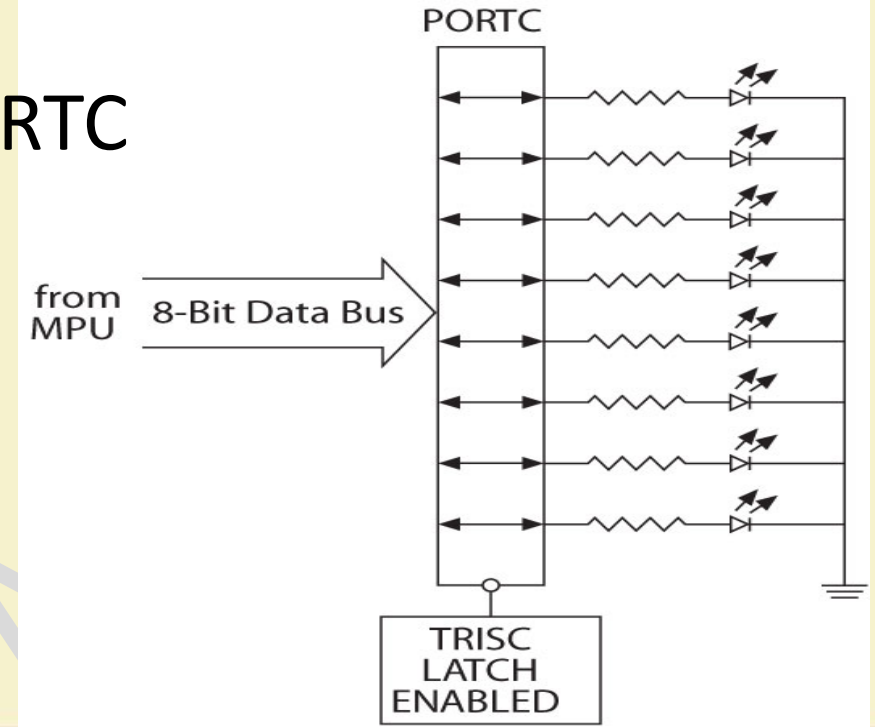
Illustrative Program

- Problem statement:
  - Write instructions to light up alternate LEDs at PORTC
- Hardware:
  - PORTC
    - Bidirectional (input or output) port
    - Setup as output port for display
  - Logic 1 will turn on an LED

NPTEL

Illustration

- Interfacing LEDs to PORTC



Illustration

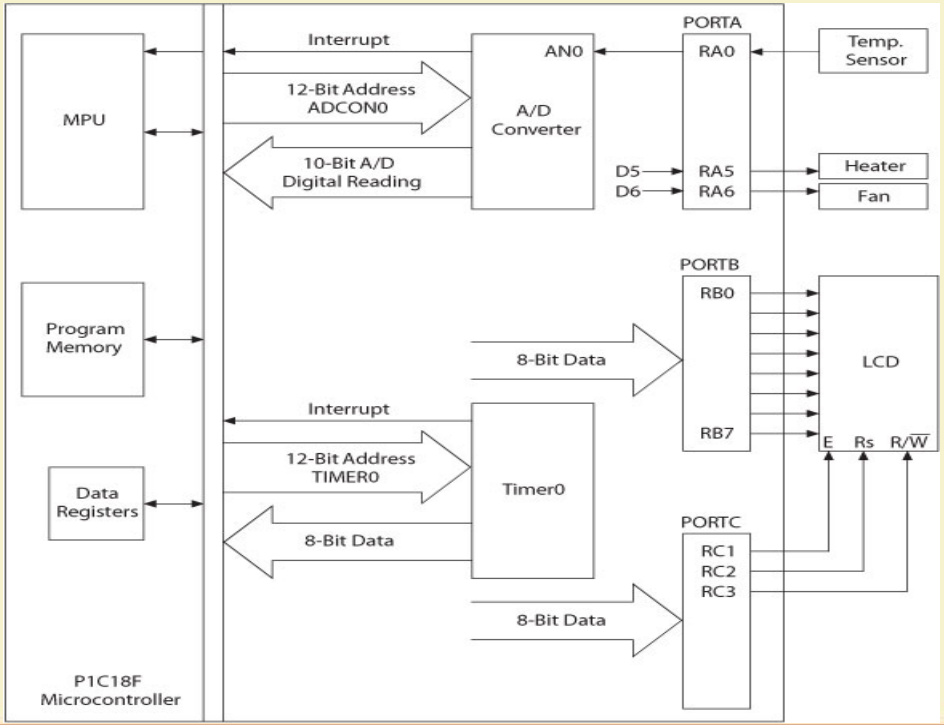
- Program (software)
  - Logic 0 to TRISC sets up PORTC as an output port
  - Byte 55_H turns on alternate LEDs

```
MOVLW   00      ;Load W register with 0
MOVWF   TRISC   ;Set up PORTC as output
MOVLW   0x55    ;Byte 55H to turn on LEDS
MOVWF   PORTC   ;Turn on LEDs
SLEEP           ;Power down
```

NPTEL

Embedded System

- MCU-based System



AVR Microcontrollers

Santanu Chattopadhyay
Electronics and Electrical Communication Engineering

NPTEL

AVR Microcontroller

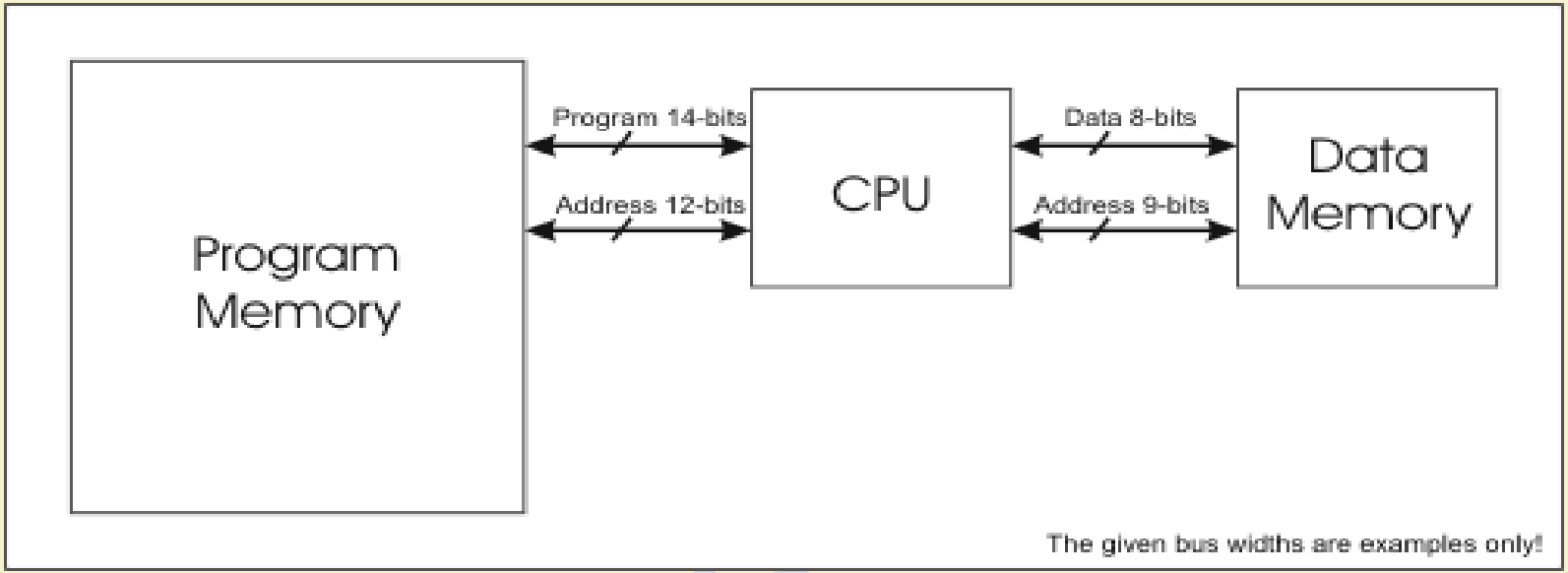
AVR stand for?

Advanced Virtual RISC,
the founders are Alf Egil Bogen Vegard Wollan RISC

AVR architecture was conceived by two students at Norwegian Institute of Technology (NTH) and further refined and developed at **Atmel Norway**, the Atmel company founded by the two chip architects.

NPTEL

AVR Microcontroller



Harvard Architecture diagram

AVR Microcontroller

A series of 8-bit RISC microcontrollers from Atmel.

All AVR microcontrollers share same instruction set and a basic CPU (Harvard) architecture.

It has 32 8-Bit general purpose registers.

Mostly instruction Execute in Single clock cycle. Which makes it faster among 8 bit microcontrollers.

AVR was designed for efficient execution of compiled C code.

NPTEL

AVR Microcontroller

AVR is a family of 8-bit microntrollers with a large range of variants differing in:

- size of program-memory (flash)
- size of EEPROM memory
- number of I/O pins
- number of on-chip features such as uart and adc

Smallest microconroller is the ATTiny11 with 1k flash ROM, no RAM and 6 I/O pins.
Large such as the ATMEGA128 with 128k flash, 4KB RAM, 53 I/O pins and lots of on-chip features.

NPTEL

Part Number	Pins	Flash	EEPROM	RAM
90S1200	20	1K	64 Bytes	0
90S2313	20	2K	128	128
90S2323	8	2K	128	128
90S2333	28	2K	128	128
90S4433	28	4K	256	128
90S4414	40	4K	256	256
90S8515	40	8K	512	512
90S4434	40	4K	256	256
90S2343	8	2K	128	128
Mega103	64	128K	4096	4096
Mega603	64	64K	2048	4096
Tiny10	8	1K	64	0
Tiny12	8	1K	64	0
Tiny13	8	2K	128	128

AVR AT90S2313 Microcontrollers

This is a microcontroller of AVR series from Atmel.

High-performance and Low-power RISC Architecture

It is a low voltage (2.7V - 6V),
high performance CMOS 8-bit micro controller based on the AVR **RISC architecture**

Since it is a microcontroller from AVR series ,it is also using Harvard Architecture

NPTEL

AVR AT90S2313 Architecture

AT90S2313 provides the following features:

- 2K bytes of In-System Programmable Flash
- 28 bytes EEPROM
- 128 bytes SRAM
- 15 general purpose I/O lines
- 32 general purpose working registers
- flexible Timer/Counters with compare modes
- internal and external interrupts
- A programmable serial UART
- one 8 bit timer/counter
- one16-bit timer/counter
- Analog Comparator
- on chip oscillator and clock circuitry

NPTEL

AVR Architecture

- **Registers**
- **Instruction Set**
- **I/O ports**
- **Memory (flash & RAM & ROM)**
- **CPU**

NPTEL

AVR Architecture

Registers: Two types of registers

GENERAL purpose & SPECIAL purpose registers

GENERAL purpose

32 general purpose registers having storage capacity of 8-Bits
Named as R0,R1,R2 to R31.
Register 0 to 15 & 16 to 31 are different.
Can store both Data & Addresses.

SPECIAL purpose: Three registers - Program counter, Stack Pointer, Status Register

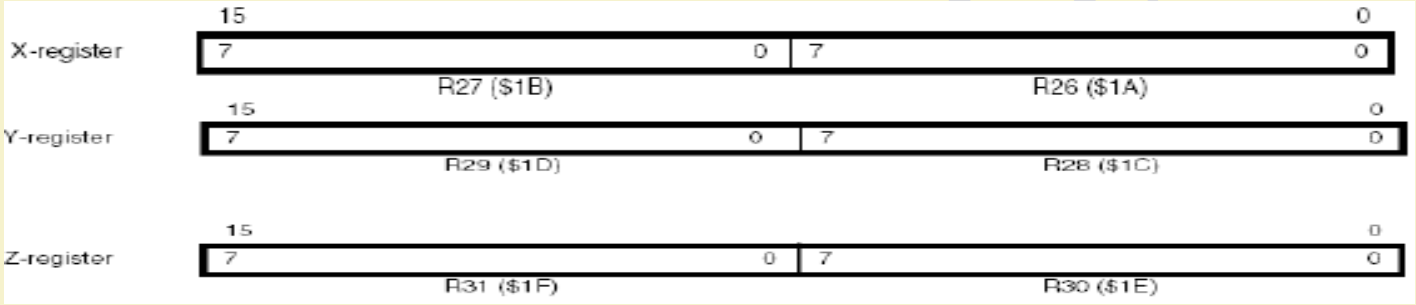
NPTEL

AVR Architecture

Pointer Register

Three 16-bit address registers pairs of registers 26 to 31 have extra meaning in AVR assembly.
X (r27:r26), y (r29:r28), z (r31:r30).

pointer	Sequence
X	**Read / Write from address X, don't change the pointer**



AVR Architecture

status register (SREG) that contains
It is 8-bit long each bit has a different meaning.

I	T	H	S	V	N	Z	C

I: Global Interrupt Enable/Disable Flag, SREG7
T: Transfer bit used by BLD and BST instructions, SREG6
H: Half Carry Flag, SREG5
S: For signed tests Instruction Set, SREG4
V: Two's complement overflow indicator, SREG3
N: Negative Flag, SREG2
Z: Zero Flag, SREG1
C: Carry Flag, SREG0

NPTEL

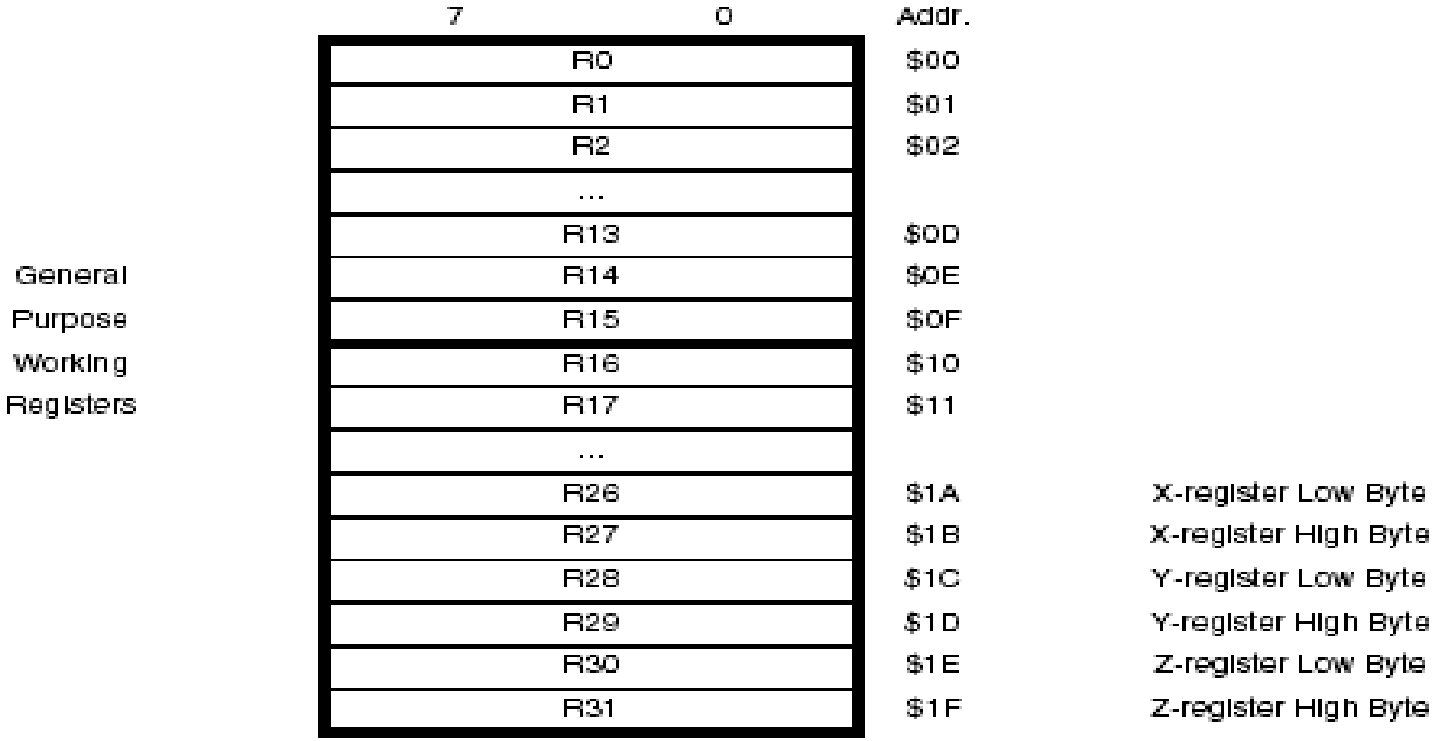
AVR Architecture

Stack Pointer (SP)

16-bit stack pointer (SP) holds address in data space of area to save function call information.

NPTEL

AVR Register Architecture



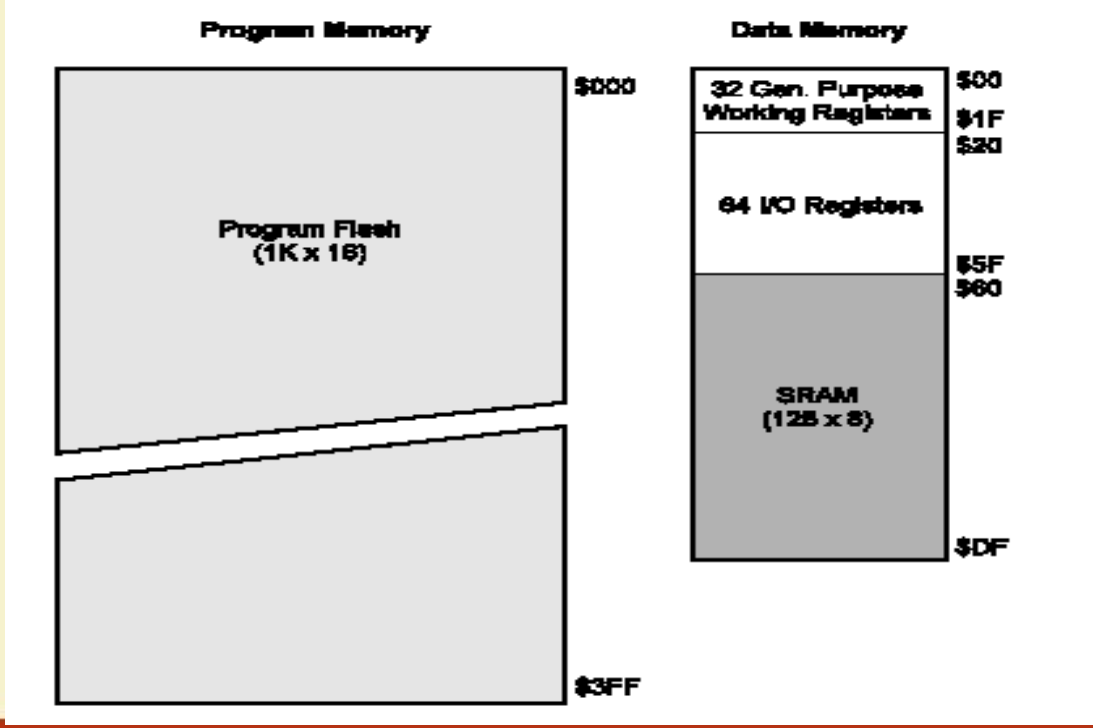
AVR Architecture

Memory:

There are two separate memories

Program Memory (Flash Memory)

Data Memory



AVR AT90S2313 Memory Architecture

Memory:

Program Memory (Flash Memory)

2K Bytes of flash memory

128 Bytes of In-System Programmable EEPROM program memory holds interrupt function addresses, 16 bit and double word (32 bit) opcode, and static data tables

NPTEL

AVR AT90S2313 Memory Architecture

Data Memory

Used for data and is separate from the program memory.

128 Bytes of SRAM

- Registers reassigned the 32 Data Space addresses ($00 - $1F)
- I/O memory space contains 64 addresses for CPU peripheral functions such as, control registers, Timer/Counters, A/D converters and other I/O functions.
- I/O memory can be accessed directly or as the Data Space locations those of the Register File, $20 - $5F.
- Stack is effectively allocated in the general data SRAM, and consequently the stack size is only limited by the total SRAM size and the usage of the SRAM.

NPTEL

AVR AT90S2313 instruction Architecture

AVR Instruction SET

118 Powerful Instructions – Most Single Clock Cycle Execution

All arithmetic operations are done on registers R0 - R31
Mostly instructions take one cycle for execution

ADD Rd,Rr
Rd: Destination (and source) register in the Register File
Rr: Source register in the Register File

NPTEL

AVR AT90S2313 instruction Architecture

Instruction ADD R23, R11 is *encoded as the 16-bit opcode 0x0EEB.*

Bit pattern : 0000 1110 1110 1011

Three components.

5 blue bits 00011 distinguish this as an *add* instruction.

5 red bits 10111 indicates *register 23* is the first operand register.

The 5 green bits 01011 indicates *register 11* is the second operand register.

All add Rd, Rr instructions follow this pattern.

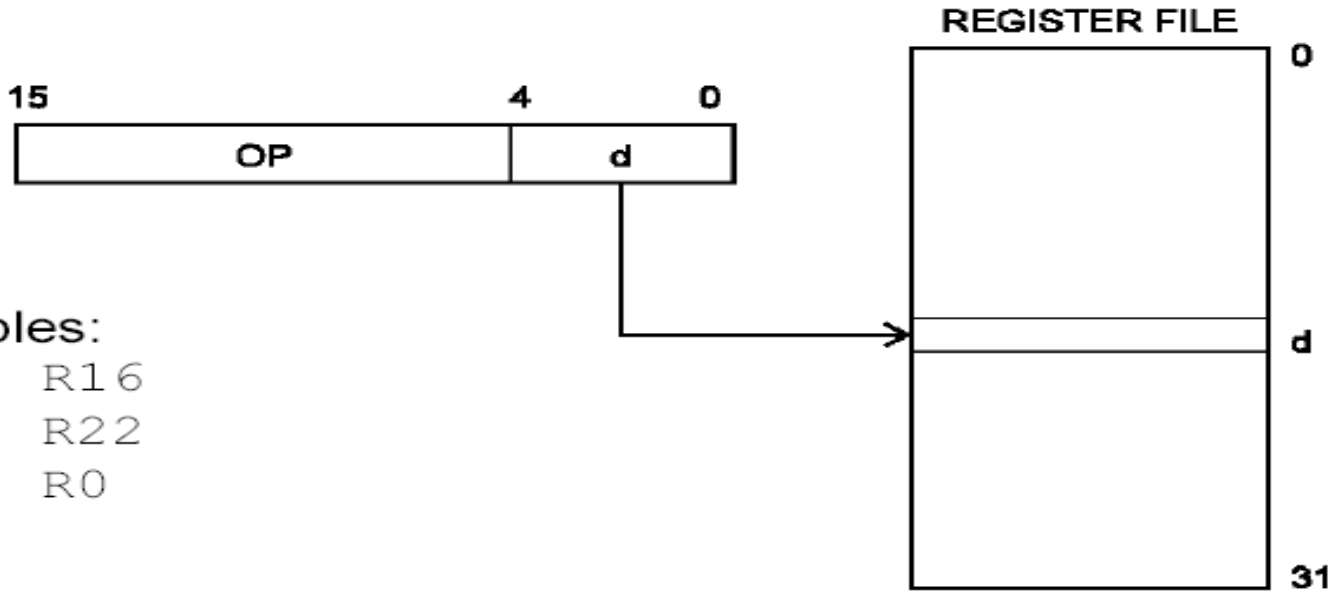
NPTEL

AVR Addressing Modes

- Register Direct, with 1 and 2 registers
- I/O Direct
- Data Direct
- Data Indirect
  - with pre-decrement
  - with post-increment
- Code Memory Addressing

NPTEL

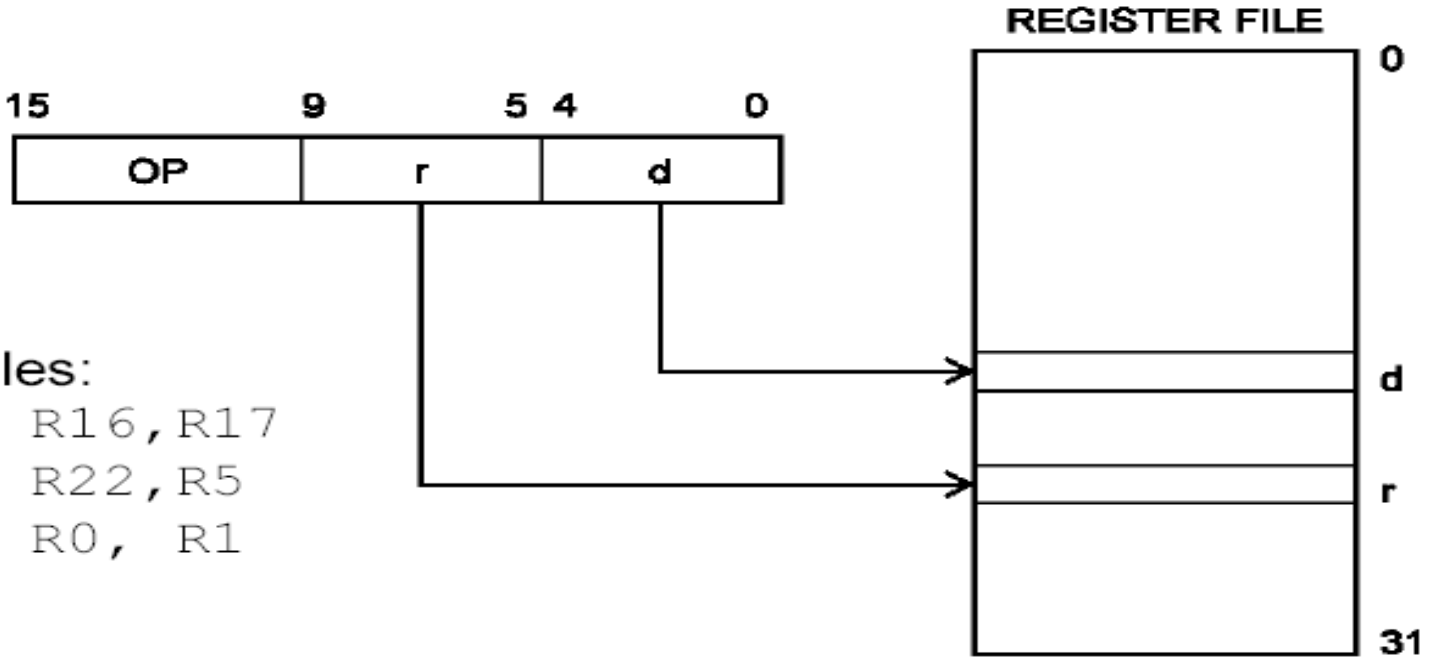
Register Direct: 1 Register



Examples:

```
INC    R16
CLR    R22
EOR    R0
```

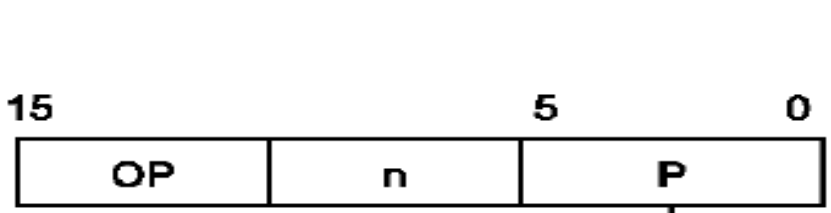
Register Direct: 2 Registers

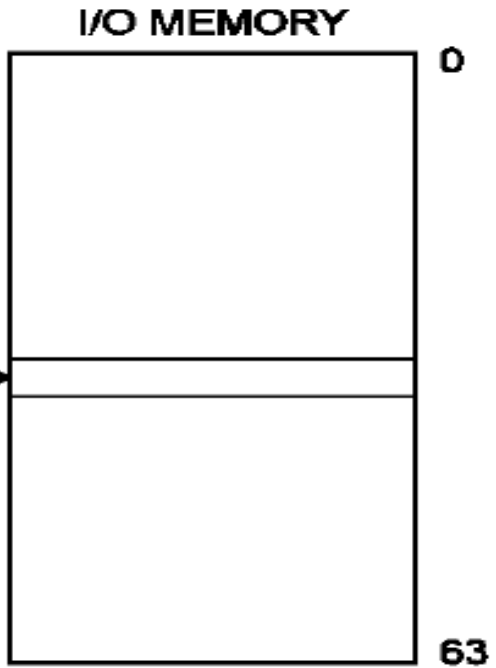


Examples:

```
ADD     R16,R17
CP      R22,R5
MOV     R0, R1
```

I/O Direct

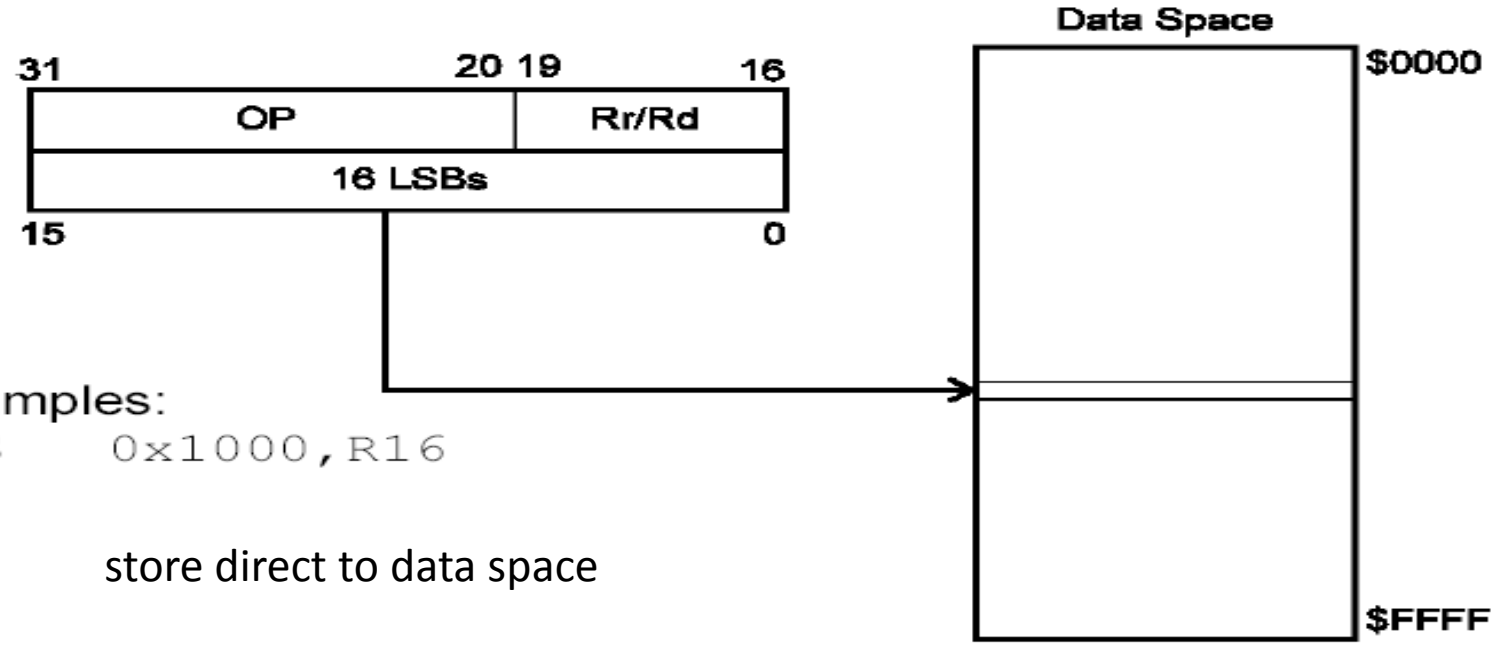




Examples:

```
IN      R16,PIND
OUT     PORTC,R16
```

Data Direct

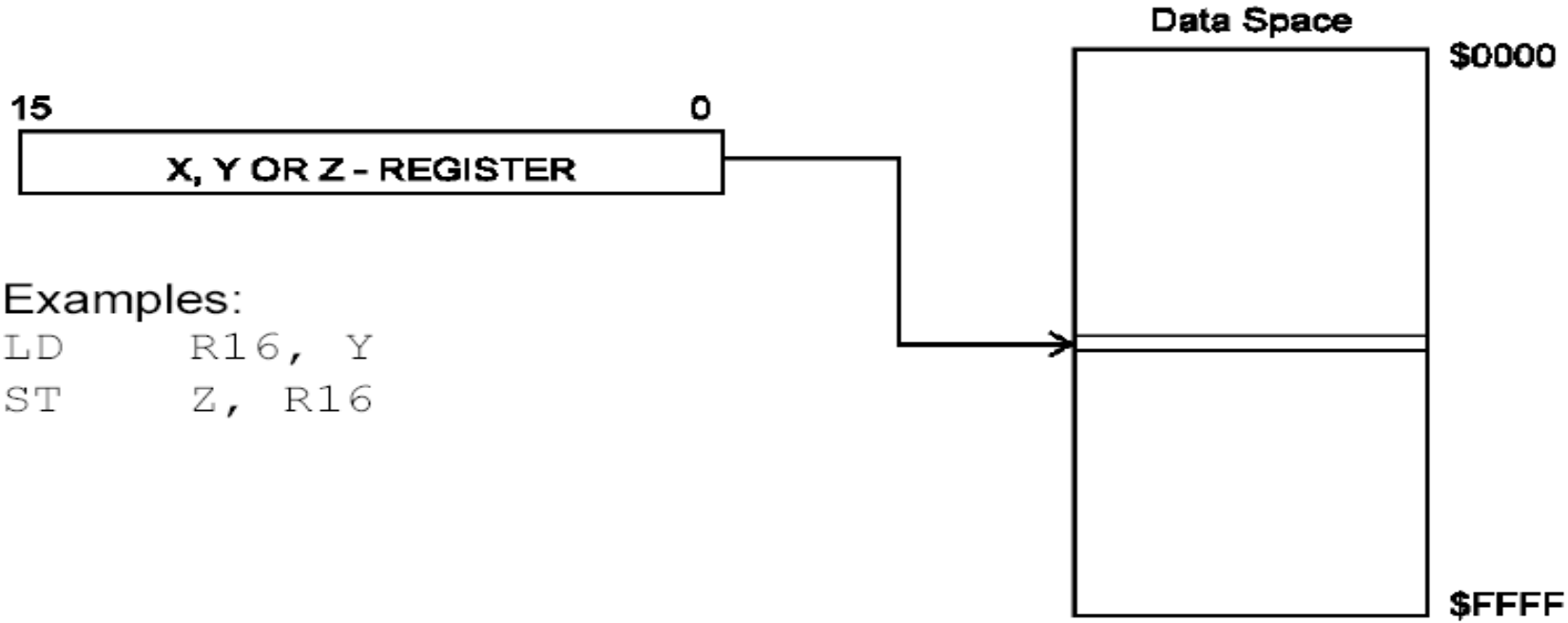


Examples:

```
STS    0x1000,R16
```

STS store direct to data space

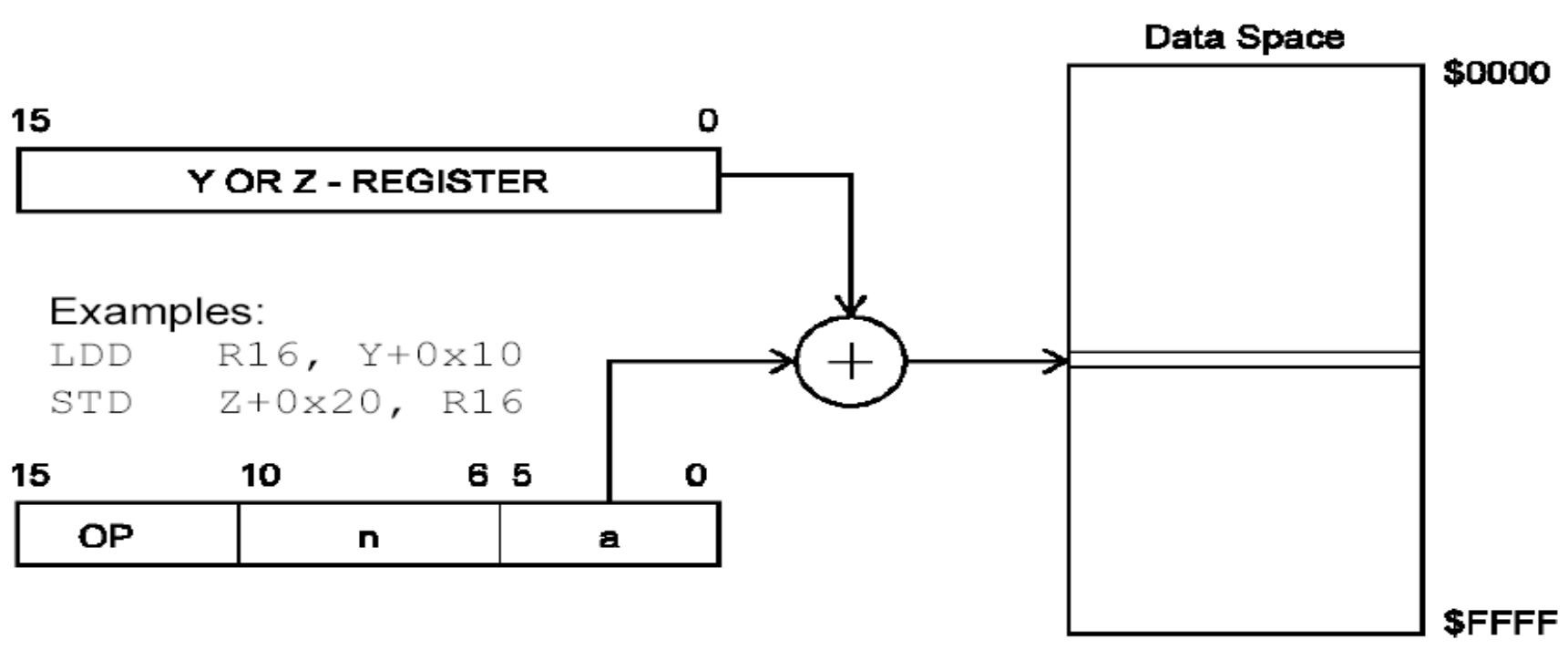
Data Indirect



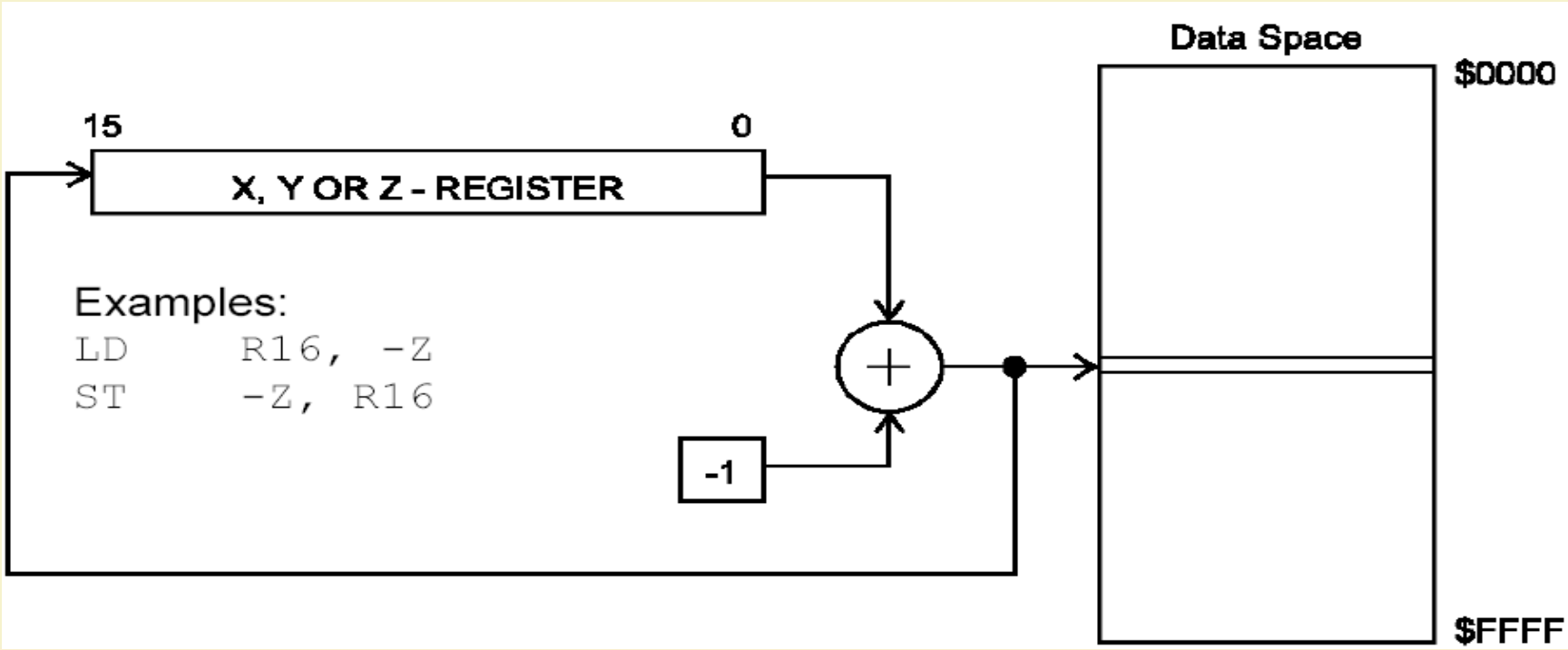
Examples:

```
LD      R16, Y
ST      Z, R16
```

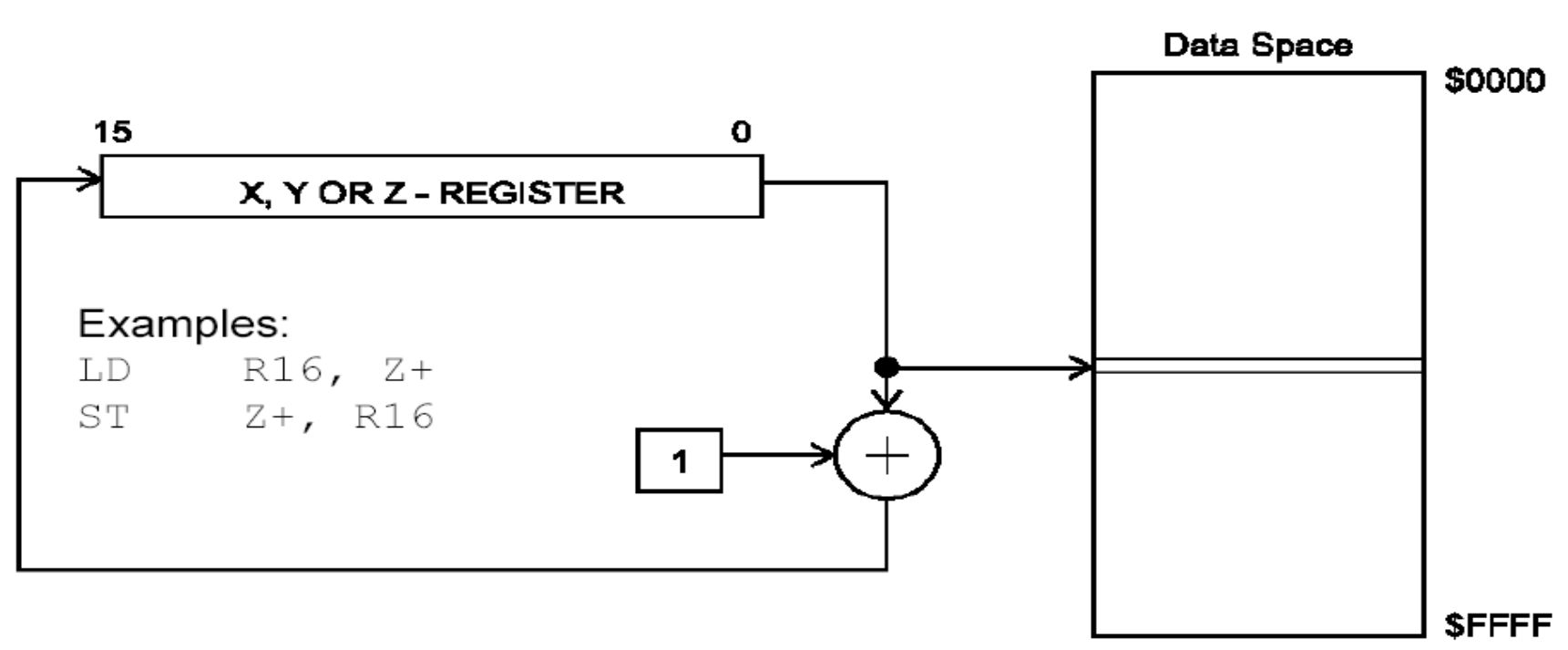
Data Indirect w/ Displacement



Data Indirect: Pre-Decrement



Data Indirect: Post-Increment



Some Interesting Instructions

- NOP – Do nothing for 1 cycle
- SLEEP – Sleep until reset or interrupted
- WDR – Watch Dog Reset

NPTEL

AVR AT90S2313 I/O Pins

General Purpose I/O Ports

Each port has 3 control registers associated with it,
DDRx, PORTx, and PINx

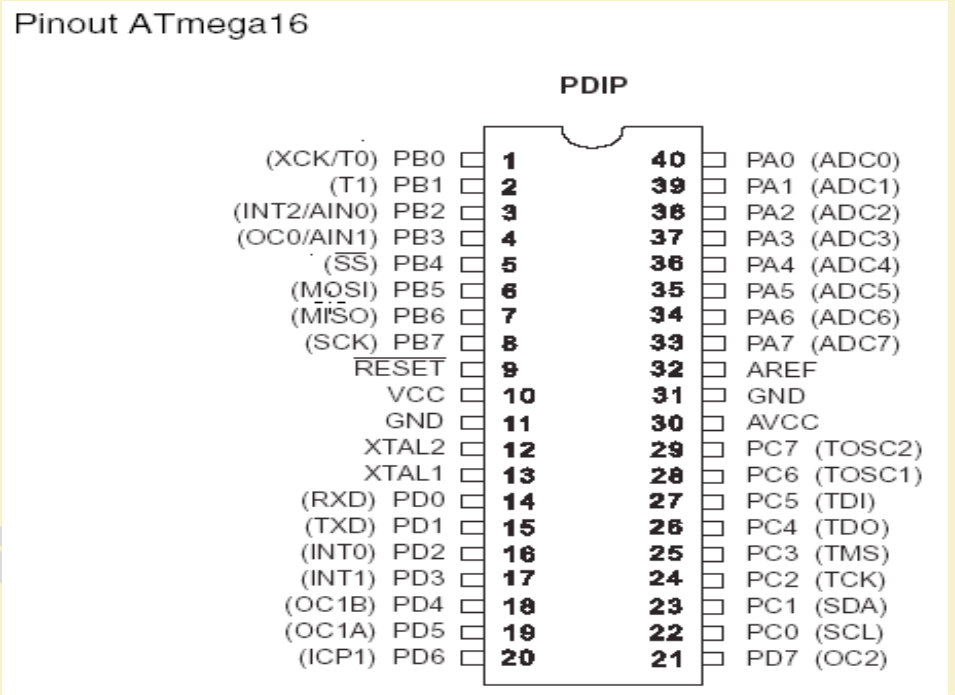
The DDR (Data Direction Register) bit tells a leg to act as an input (0), or output (1).
The PORT (Pin Output / Read Tweak)
The PIN (Port INput) register is read only,

I/O and Packages
– 15 Programmable I/O Lines

NPTEL

AVR Timer/Counter 0

- 8 Bit
- Wrap-Around Up Counter
- Interrupt on overflow



AVR Timer/Counter 0

- 8 Bit Up Counter
  - counts from 0 to 255 (0xFF), then loops to 0
  - Internal or External Clock source
- Prescaler
- Output capture through OC0, i.e. PB3, pin 4
- Interrupt on Overflow
  - Transition from 255 to 0 can trigger interrupt if desired

NPTEL

AVR Timer/Counter 0

OC0, Output Compare Match output:

Whenever TCNT0 equals OCR0 (Output Compare Register 0), the comparator signals a match

The PB3 pin can serve as an external output for the Timer/Counter0 Compare Match. The PB3 pin has to be configured as an output

NPTEL

AVR Timer/Counter 1

- 16 Bit
- Dual Comparators A,B (output compares)
- Up Counter
- Interrupt on:
  - Overflow
  - Compare A/B
  - Input Capture of external event on ICP pin.
- Can also act as an 8, 9 or 10 bit PWM Up-Down Counter.

NPTEL

AVR Timer/Counter 1

•The Input Capture unit of Timer/Counter captures external events and gives them a time-stamp indicating time of occurrence.

•The external signal indicating an event, or multiple events, can be applied via the ICP1 pin or alternatively, via the Analog Comparator unit.

•The time-stamps can then be used to calculate frequency, duty-cycle, and other features of the signal applied.

•Alternatively the time-stamps can be used for creating a log of the events.

NPTEL

Timer 1 and Output Compare

- The AVR has two *output compares* (OCR1A/B)
  - OCR1A/B are 16-bit registers
  - When the value of OCR1A/OCR1B matches that of Timer1:
    - A user-defined action can take place on the OC1A/OC1B pin (set/clear/inv)

    i.e.,PD5 /PD4 need to set as output
    - An interrupt can be triggered
    - Timer1 can be cleared to zero
  - Once set up, output compares operate continuously without software intervention
  - Great for:
    - Precise recurring timing
    - Frequency/Tone generation (maybe sound effects)
    - All kinds of digital signal generation
      - Infrared communications
      - Software-driven serial ports

NPTEL