

Microcomputer Engineering

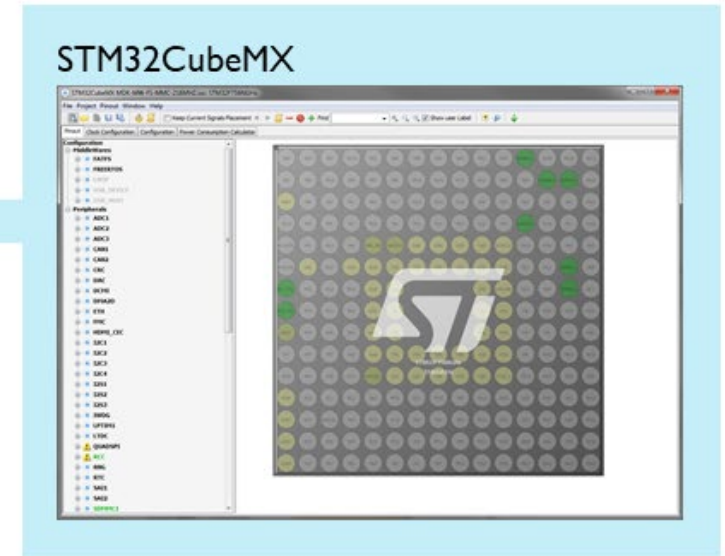
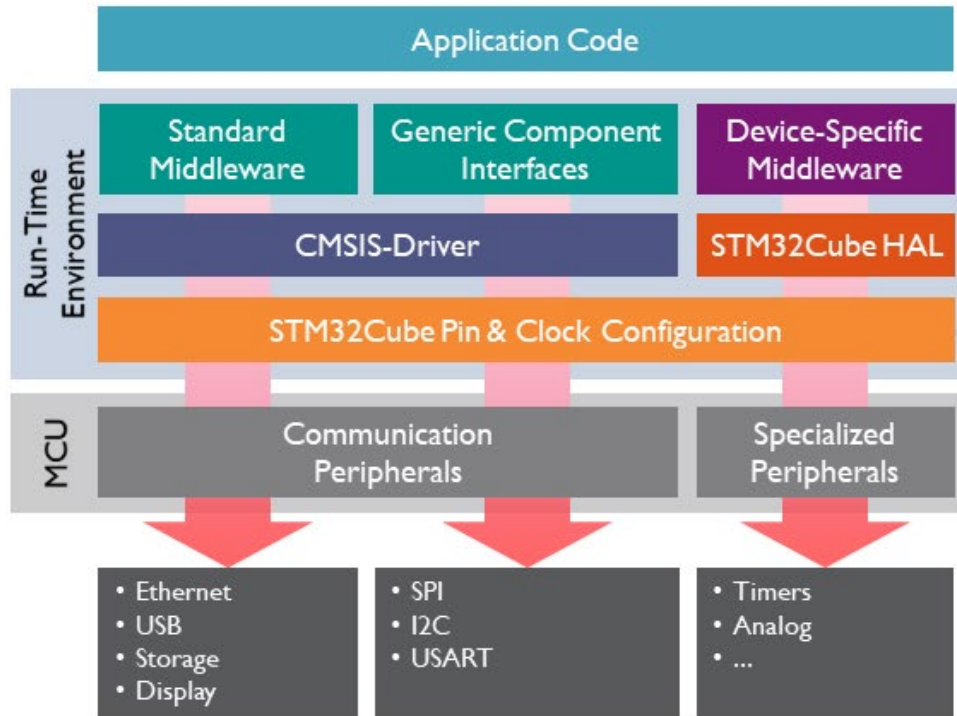
TMIK13

Lecture 3

FINITE STATE MACHINES (FSM)

ANDREAS AXELSSON (ANDREAS.AXELSSON@JU.SE)

Microchip – Abstraction Layers



STM32CubeMX – Peripherals

STM32CubeMX interface showing Pinout & Configuration, Clock Configuration, Project Manager, and Tools tabs. The Pinout view is active, displaying the STM32F401RETx LQFP64 pinout and the GPIO Mode and Configuration table.

GPIO Mode and Configuration

Configuration: Group By Peripherals

Selected Peripherals: SYS, USART, NVIC, GPIO, Single Mapped Signals, RCC

Search Signals: Search (Ctrl+F) ☐ Show only Modified Pins

Pin	Signal	GPIO	GPIO	GPIO	Maxim	User L...	Modified
PA4	n/a	Low	Output ...	No pull...	Low		
PA5	n/a	Low	Output ...	No pull...	Low	LD2 [G...	<input checked="" type="checkbox"/>
PA6	n/a	Low	Output ...	No pull...	Low		<input type="checkbox"/>
PA7	n/a	Low	Output ...	No pull...	Low		<input type="checkbox"/>
PA8	n/a	Low	Output ...	No pull...	Low		<input type="checkbox"/>
PA9	n/a	Low	Output ...	No pull...	Low		<input type="checkbox"/>
PA10	n/a	Low	Output ...	No pull...	Low		<input type="checkbox"/>
PC13...	n/a	n/a	Extern...	No pull...	n/a	B1 [Blu...	<input checked="" type="checkbox"/>

Select Pins from table to configure them. Multiple selection is Allowed.

Pinout view

STM32F401RETx LQFP64

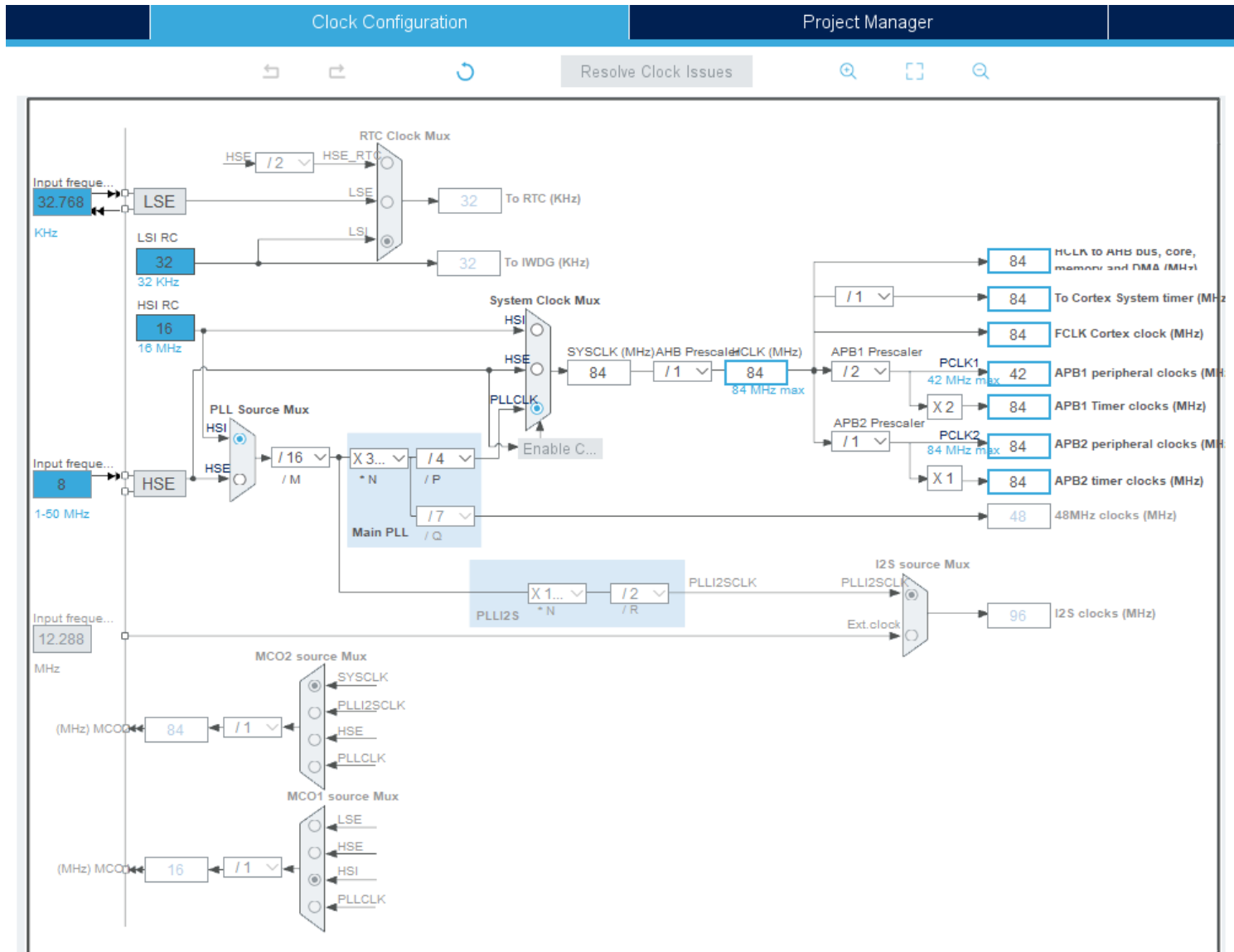
Pinout details (Left side):

- VBAT
- PC13...
- RCC_OSC32_IN
- RCC_OSC32_OUT
- RCC_OSC_IN
- RCC_OSC_OUT
- NRST
- PC0
- PC1
- PC2
- PC3
- VSSA..
- VREF+
- PA0...
- PA1
- USART_TX
- PA2
- PA3
- VSS
- VDD
- PA4
- PA5
- PA6
- PA7
- PC4
- PC5
- PC6
- PC7
- PC8
- PC9
- PC10
- PC11
- PC12
- PC13
- PC14
- PC15
- PC16
- PC17
- PC18
- PC19
- PC20
- PC21
- PC22
- PC23
- PC24
- PC25
- PC26
- PC27
- PC28
- PC29
- PC30
- PC31
- PC32
- PC33
- PC34
- PC35
- PC36
- PC37
- PC38
- PC39
- PC40
- PC41
- PC42
- PC43
- PC44
- PC45
- PC46
- PC47
- PC48
- PC49
- PC50
- PC51
- PC52
- PC53
- PC54
- PC55
- PC56
- PC57
- PC58
- PC59
- PC60
- PC61
- PC62
- PC63
- PC64
- PC65
- PC66
- PC67
- PC68
- PC69
- PC70
- PC71
- PC72
- PC73
- PC74
- PC75
- PC76
- PC77
- PC78
- PC79
- PC80
- PC81
- PC82
- PC83
- PC84
- PC85
- PC86
- PC87
- PC88
- PC89
- PC90
- PC91
- PC92
- PC93
- PC94
- PC95
- PC96
- PC97
- PC98
- PC99
- PC100
- PC101
- PC102
- PC103
- PC104
- PC105
- PC106
- PC107
- PC108
- PC109
- PC110
- PC111
- PC112
- PC113
- PC114
- PC115
- PC116
- PC117
- PC118
- PC119
- PC120
- PC121
- PC122
- PC123
- PC124
- PC125
- PC126
- PC127
- PC128
- PC129
- PC130
- PC131
- PC132
- PC133
- PC134
- PC135
- PC136
- PC137
- PC138
- PC139
- PC140
- PC141
- PC142
- PC143
- PC144
- PC145
- PC146
- PC147
- PC148
- PC149
- PC150
- PC151
- PC152
- PC153
- PC154
- PC155
- PC156
- PC157
- PC158
- PC159
- PC160
- PC161
- PC162
- PC163
- PC164
- PC165
- PC166
- PC167
- PC168
- PC169
- PC170
- PC171
- PC172
- PC173
- PC174
- PC175
- PC176
- PC177
- PC178
- PC179
- PC180
- PC181
- PC182
- PC183
- PC184
- PC185
- PC186
- PC187
- PC188
- PC189
- PC190
- PC191
- PC192
- PC193
- PC194
- PC195
- PC196
- PC197
- PC198
- PC199
- PC200
- PC201
- PC202
- PC203
- PC204
- PC205
- PC206
- PC207
- PC208
- PC209
- PC210
- PC211
- PC212
- PC213
- PC214
- PC215
- PC216
- PC217
- PC218
- PC219
- PC220
- PC221
- PC222
- PC223
- PC224
- PC225
- PC226
- PC227
- PC228
- PC229
- PC230
- PC231
- PC232
- PC233
- PC234
- PC235
- PC236
- PC237
- PC238
- PC239
- PC240
- PC241
- PC242
- PC243
- PC244
- PC245
- PC246
- PC247
- PC248
- PC249
- PC250
- PC251
- PC252
- PC253
- PC254
- PC255
- PC256
- PC257
- PC258
- PC259
- PC260
- PC261
- PC262
- PC263
- PC264
- PC265
- PC266
- PC267
- PC268
- PC269
- PC270
- PC271
- PC272
- PC273
- PC274
- PC275
- PC276
- PC277
- PC278
- PC279
- PC280
- PC281
- PC282
- PC283
- PC284
- PC285
- PC286
- PC287
- PC288
- PC289
- PC290
- PC291
- PC292
- PC293
- PC294
- PC295
- PC296
- PC297
- PC298
- PC299
- PC300
- PC301
- PC302
- PC303
- PC304
- PC305
- PC306
- PC307
- PC308
- PC309
- PC310
- PC311
- PC312
- PC313
- PC314
- PC315
- PC316
- PC317
- PC318
- PC319
- PC320
- PC321
- PC322
- PC323
- PC324
- PC325
- PC326
- PC327
- PC328
- PC329
- PC330
- PC331
- PC332
- PC333
- PC334
- PC335
- PC336
- PC337
- PC338
- PC339
- PC340
- PC341
- PC342
- PC343
- PC344
- PC345
- PC346
- PC347
- PC348
- PC349
- PC350
- PC351
- PC352
- PC353
- PC354
- PC355
- PC356
- PC357
- PC358
- PC359
- PC360
- PC361
- PC362
- PC363
- PC364
- PC365
- PC366
- PC367
- PC368
- PC369
- PC370
- PC371
- PC372
- PC373
- PC374
- PC375
- PC376
- PC377
- PC378
- PC379
- PC380
- PC381
- PC382
- PC383
- PC384
- PC385
- PC386
- PC387
- PC388
- PC389
- PC390
- PC391
- PC392
- PC393
- PC394
- PC395
- PC396
- PC397
- PC398
- PC399
- PC400
- PC401
- PC402
- PC403
- PC404
- PC405
- PC406
- PC407
- PC408
- PC409
- PC410
- PC411
- PC412
- PC413
- PC414
- PC415
- PC416
- PC417
- PC418
- PC419
- PC420
- PC421
- PC422
- PC423
- PC424
- PC425
- PC426
- PC427
- PC428
- PC429
- PC430
- PC431
- PC432
- PC433
- PC434
- PC435
- PC436
- PC437
- PC438
- PC439
- PC440
- PC441
- PC442
- PC443
- PC444
- PC445
- PC446
- PC447
- PC448
- PC449
- PC450
- PC451
- PC452
- PC453
- PC454
- PC455
- PC456
- PC457
- PC458
- PC459
- PC460
- PC461
- PC462
- PC463
- PC464
- PC465
- PC466
- PC467
- PC468
- PC469
- PC470
- PC471
- PC472
- PC473
- PC474
- PC475
- PC476
- PC477
- PC478
- PC479
- PC480
- PC481
- PC482
- PC483
- PC484
- PC485
- PC486
- PC487
- PC488
- PC489
- PC490
- PC491
- PC492
- PC493
- PC494
- PC495
- PC496
- PC497
- PC498
- PC499
- PC500
- PC501
- PC502
- PC503
- PC504
- PC505
- PC506
- PC507
- PC508
- PC509
- PC510
- PC511
- PC512
- PC513
- PC514
- PC515
- PC516
- PC517
- PC518
- PC519
- PC520
- PC521
- PC522
- PC523
- PC524
- PC525
- PC526
- PC527
- PC528
- PC529
- PC530
- PC531
- PC532
- PC533
- PC534
- PC535
- PC536
- PC537
- PC538
- PC539
- PC540
- PC541
- PC542
- PC543
- PC544
- PC545
- PC546
- PC547
- PC548
- PC549
- PC550
- PC551
- PC552
- PC553
- PC554
- PC555
- PC556
- PC557
- PC558
- PC559
- PC560
- PC561
- PC562
- PC563
- PC564
- PC565
- PC566
- PC567
- PC568
- PC569
- PC570
- PC571
- PC572
- PC573
- PC574
- PC575
- PC576
- PC577
- PC578
- PC579
- PC580
- PC581
- PC582
- PC583
- PC584
- PC585
- PC586
- PC587
- PC588
- PC589
- PC590
- PC591
- PC592
- PC593
- PC594
- PC595
- PC596
- PC597
- PC598
- PC599
- PC600
- PC601
- PC602
- PC603
- PC604
- PC605
- PC606
- PC607
- PC608
- PC609
- PC610
- PC611
- PC612
- PC613
- PC614
- PC615
- PC616
- PC617
- PC618
- PC619
- PC620
- PC621
- PC622
- PC623
- PC624
- PC625
- PC626
- PC627
- PC628
- PC629
- PC630
- PC631
- PC632
- PC633
- PC634
- PC635
- PC636
- PC637
- PC638
- PC639
- PC640
- PC641
- PC642
- PC643
- PC644
- PC645
- PC646
- PC647
- PC648
- PC649
- PC650
- PC651
- PC652
- PC653
- PC654
- PC655
- PC656
- PC657
- PC658
- PC659
- PC660
- PC661
- PC662
- PC663
- PC664
- PC665
- PC666
- PC667
- PC668
- PC669
- PC670
- PC671
- PC672
- PC673
- PC674
- PC675
- PC676
- PC677
- PC678
- PC679
- PC680
- PC681
- PC682
- PC683
- PC684
- PC685
- PC686
- PC687
- PC688
- PC689
- PC690
- PC691
- PC692
- PC693
- PC694
- PC695
- PC696
- PC697
- PC698
- PC699
- PC700
- PC701
- PC702
- PC703
- PC704
- PC705
- PC706
- PC707
- PC708
- PC709
- PC710
- PC711
- PC712
- PC713
- PC714
- PC715
- PC716
- PC717
- PC718
- PC719
- PC720
- PC721
- PC722
- PC723
- PC724
- PC725
- PC726
- PC727
- PC728
- PC729
- PC730
- PC731
- PC732
- PC733
- PC734
- PC735
- PC736
- PC737
- PC738
- PC739
- PC740
- PC741
- PC742
- PC743
- PC744
- PC745
- PC746
- PC747
- PC748
- PC749
- PC750
- PC751
- PC752
- PC753
- PC754
- PC755
- PC756
- PC757
- PC758
- PC759
- PC760
- PC761
- PC762
- PC763
- PC764
- PC765
- PC766
- PC767
- PC768
- PC769
- PC770
- PC771
- PC772
- PC773
- PC774
- PC775
- PC776
- PC777
- PC778
- PC779
- PC780
- PC781
- PC782
- PC783
- PC784
- PC785
- PC786
- PC787
- PC788
- PC789
- PC790
- PC791
- PC792
- PC793
- PC794
- PC795
- PC796
- PC797
- PC798
- PC799
- PC800
- PC801
- PC802
- PC803
- PC804
- PC805
- PC806
- PC807
- PC808
- PC809
- PC810
- PC811
- PC812
- PC813
- PC814
- PC815
- PC816
- PC817
- PC818
- PC819
- PC820
- PC821
- PC822
- PC823
- PC824
- PC825
- PC826
- PC827
- PC828
- PC829
- PC830
- PC831
- PC832
- PC833
- PC834
- PC835
- PC836
- PC837
- PC838
- PC839
- PC840
- PC841
- PC842
- PC843
- PC844
- PC845
- PC846
- PC847
- PC848
- PC849
- PC850
- PC851
- PC852
- PC853
- PC854
- PC855
- PC856
- PC857
- PC858
- PC859
- PC860
- PC861
- PC862
- PC863
- PC864
- PC865
- PC866
- PC867
- PC868
- PC869
- PC870
- PC871
- PC872
- PC873
- PC874
- PC875
- PC876
- PC877
- PC878
- PC879
- PC880
- PC881
- PC882
- PC883
- PC884
- PC885
- PC886
- PC887
- PC888
- PC889
- PC890
- PC891
- PC892
- PC893
- PC894
- PC895
- PC896
- PC897
- PC898
- PC899
- PC900
- PC901
- PC902
- PC903
- PC904
- PC905
- PC906
- PC907
- PC908
- PC909
- PC910
- PC911
- PC912
- PC913
- PC914
- PC915
- PC916
- PC917
- PC918
- PC919
- PC920
- PC921
- PC922
- PC923
- PC924
- PC925
- PC926
- PC927
- PC928
- PC929
- PC930
- PC931
- PC932
- PC933
- PC934
- PC935
- PC936
- PC937
- PC938
- PC939
- PC940
- PC941
- PC942
- PC943
- PC944
- PC945
- PC946
- PC947
- PC948
- PC949
- PC950
- PC951
- PC952
- PC953
- PC954
- PC955
- PC956
- PC957
- PC958
- PC959
- PC960
- PC961
- PC962
- PC963
- PC964
- PC965
- PC966
- PC967
- PC968
- PC969
- PC970
- PC971
- PC972
- PC973
- PC974
- PC975
- PC976
- PC977
- PC978
- PC979
- PC980
- PC981
- PC982
- PC983
- PC984
- PC985
- PC986
- PC987
- PC988
- PC989
- PC990
- PC991
- PC992
- PC993
- PC994
- PC995
- PC996
- PC997
- PC998
- PC999
- PC1000

Pinout details (Right side):

- VDD
- VSS
- PA13
- PA12
- PA11
- PA10
- PA9
- PA8
- PC9
- PC8
- PC7
- PC6
- PB15
- PB14
- PB13
- PB12
- PC5
- PC4
- PC3
- PC2
- PC1
- PC0
- NRST
- PH1
- RCC_OSC_OUT
- RCC_OSC_IN
- RCC_OSC32_OUT
- RCC_OSC32_IN
- PC13
- PC14
- PC15
- PC16
- PC17
- PC18
- PC19
- PC20
- PC21
- PC22
- PC23
- PC24
- PC25
- PC26
- PC27
- PC28
- PC29
- PC30
- PC31
- PC32
- PC33
- PC34
- PC35
- PC36
- PC37
- PC38
- PC39
- PC40
- PC41
- PC42
- PC43
- PC44
- PC45
- PC46
- PC47
- PC48
- PC49
- PC50
- PC51
- PC52
- PC53
- PC54
- PC55
- PC56
- PC57
- PC58
- PC59
- PC60
- PC61
- PC62
- PC63
- PC64
- PC65
- PC66
- PC67
- PC68
- PC69
- PC70
- PC71
- PC72
- PC73
- PC74
- PC75
- PC76
- PC77
- PC78
- PC79
- PC80
- PC81
- PC82
- PC83
- PC84
- PC85
- PC86
- PC87
- PC88
- PC89
- PC90
- PC91
- PC92
- PC93
- PC94
- PC95
- PC96
- PC97
- PC98
- PC99
- PC100
- PC101
- PC102
- PC103
- PC104
- PC105
- PC106
- PC107
- PC108
- PC109
- PC110
- PC111
- PC112
- PC113
- PC114
- PC115
- PC116
- PC117
- PC118
- PC119
- PC120
- PC121
- PC122
- PC123
- PC124
- PC125
- PC126
- PC127
- PC128
- PC129
- PC130
- PC131
- PC132
- PC133
- PC134
- PC135
- PC136
- PC137
- PC138
- PC139
- PC140
- PC141
- PC142
- PC143
- PC144
- PC145
- PC146
- PC147
- PC148
- PC149
- PC150
- PC151
- PC152
- PC153
- PC154
- PC155
- PC156
- PC157
- PC158
- PC159
- PC160
- PC161
- PC162
- PC163
- PC164
- PC165
- PC166
- PC167
- PC168
- PC169
- PC170
- PC171
- PC172
- PC173
- PC174
- PC175
- PC176
- PC177
- PC178
- PC179
- PC180
- PC181
- PC182
- PC183
- PC184
- PC185
- PC186
- PC187
- PC188
- PC189
- PC190
- PC191
- PC192
- PC193
- PC194
- PC195
- PC196
- PC197
- PC198
- PC199
- PC200
- PC201
- PC202
- PC203
- PC204
- PC205
- PC206
- PC207
- PC208
- PC209
- PC210
- PC211
- PC212
- PC213
- PC214
- PC215
- PC216
- PC217
- PC218
- PC219
- PC220
- PC221
- PC222
- PC223
- PC224
- PC225
- PC226
- PC227
- PC228
- PC229
- PC230
- PC231
- PC232
- PC233
- PC234
- PC235
- PC236
- PC237
- PC238
- PC239
- PC240
- PC241
- PC242
- PC243
- PC244
- PC245
- PC246
- PC247
- PC248
- PC249
- PC250
- PC251
- PC252
- PC253
- PC254
- PC255
- PC256
- PC257
- PC258
- PC259
- PC260
- PC261
- PC262
- PC263
- PC264
- PC265
- PC266
- PC267
- PC268
- PC269
- PC270
- PC271
- PC272
- PC273
- PC274
- PC275
- PC276
- PC277
- PC278
- PC279
- PC280

STM32CubeMX – Clock Config



STM32CubeIDE – Generated Code

STM32CubeIDE - Tarning-lab1/Core/Src/main.c - STM32CubeIDE

File Edit Source Refactor Navigate Search Project Run Percepio Window Help

Project Explorer

- IDE Blinky-lab0
- IDE Tarning-lab1
 - Includes
 - Core
 - Inc
 - main.h
 - stm32f4xx_hal_conf.h
 - stm32f4xx_it.h
 - Src
 - main.c
 - stm32f4xx_hal_msp.c
 - stm32f4xx_it.c
 - syscalls.c
 - systemem.c
 - system_stm32f4xx.c
 - Startup
 - Drivers
 - CMSIS
 - STM32F4xx_HAL_Driver
 - Inc
 - Src
 - stm32f4xx_hal_cortex.c
 - stm32f4xx_hal_dma_ex.c
 - stm32f4xx_hal_dma.c
 - stm32f4xx_hal_exti.c
 - stm32f4xx_hal_flash_ex.c
 - stm32f4xx_hal_flash_ramfunc.c

main.c

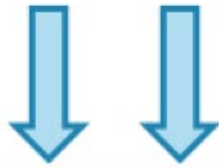
```
192  /* GPIO Ports Clock Enable */
193  __HAL_RCC_GPIOC_CLK_ENABLE();
194  __HAL_RCC_GPIOH_CLK_ENABLE();
195  __HAL_RCC_GPIOA_CLK_ENABLE();
196  __HAL_RCC_GPIOB_CLK_ENABLE();
197
198  /*Configure GPIO pin Output Level */
199  HAL_GPIO_WritePin(GPIOA, GPIO_PIN_4|LD2_Pin|GPIO_PIN_6|GPIO_PIN_7
200                      |GPIO_PIN_8|GPIO_PIN_9|GPIO_PIN_10, GPIO_P
201
202  /*Configure GPIO pin : B1_Pin */
203  GPIO_InitStruct.Pin = B1_Pin;
204  GPIO_InitStruct.Mode = GPIO_MODE_IT_FALLING;
205  GPIO_InitStruct.Pull = GPIO_NOPULL;
206  HAL_GPIO_Init(B1_GPIO_Port, &GPIO_InitStruct);
207
208  /*Configure GPIO pins : PA4 LD2_Pin PA6 PA7
209                      PA8 PA9 PA10 */
210  GPIO_InitStruct.Pin = GPIO_PIN_4|LD2_Pin|GPIO_PIN_6|GPIO_PIN_7
211                      |GPIO_PIN_8|GPIO_PIN_9|GPIO_PIN_10;
212  GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
213  GPIO_InitStruct.Pull = GPIO_NOPULL;
214  GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
215  HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);
216
217 }
218
```

Call Hierarchy

Callers of HAL_IncTick() - /Tarning-lab1/Drivers/STM32F4xx HAL Driver/Src/stm32f4xx_hal.c - in wor

STM32CubeMX

STM32 Microcontrollers
Reference Manual & Datasheets

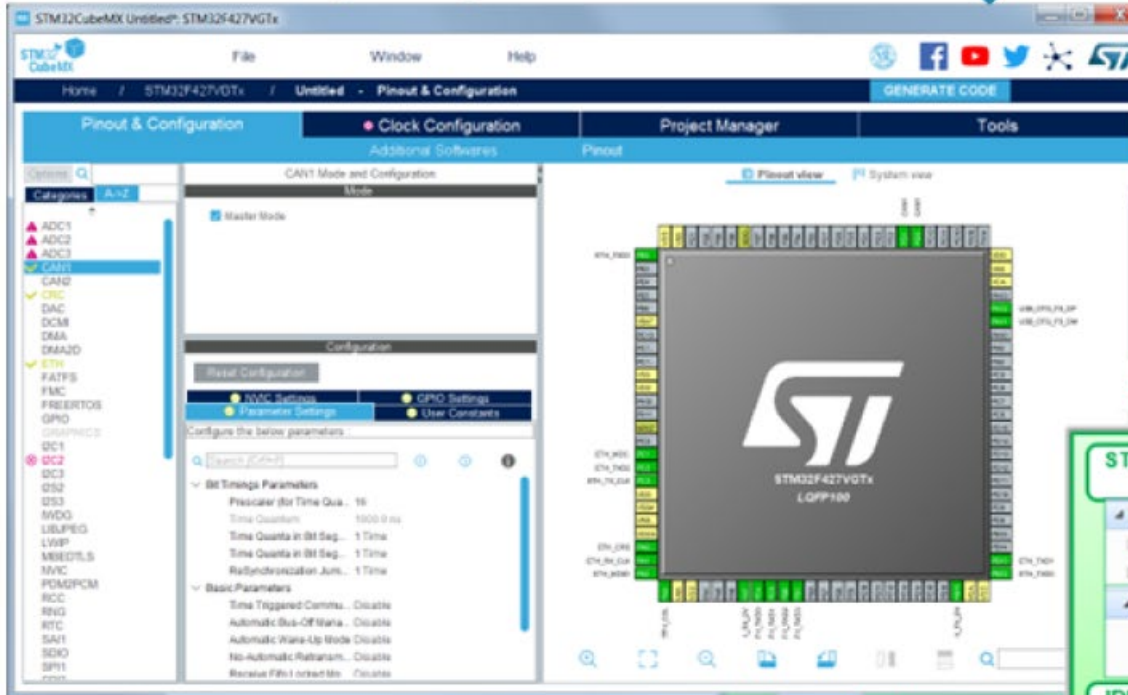


STM32CubeF4 Embedded Software Package

C projects,
Documentation, Utilities

Middleware Libraries
(freeRTOS, LwIP, USB, Graphics, etc.)

Drivers
BSP, CMSIS,
HAL(Hardware abstraction layer)
LL (Low level drivers)



STM32 Microcontroller configuration
using STM32CubeMX

STM32CubeMX generated C project

STM32CubeF4 Libraries
(Copied)

Drivers
CMSIS
STM32F4xx_HAL_Driver
Middlewares
ST
Third_Party

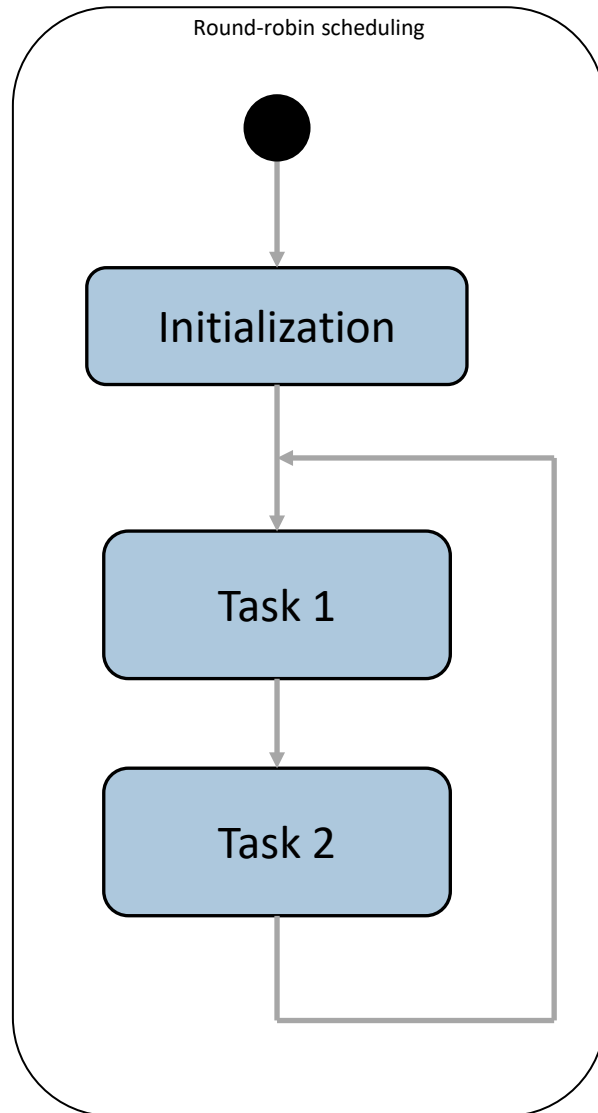
IDE Specific project files
(Generated)

EWARM
MDK-ARM
SW4STM32

Initialization Files
(Generated C Code)

ethernetif.h	ethernetif.c
lwip.h	lwip.c
lwipopts.h	main.c
main.h	stm32f4xx_hal_msp.c
stm32f4xx_hal_conf.h	stm32f4xx_it.c
stm32f4xx_it.h	system_stm32f4xx.c
usb_host.h	usb_host.c
usbh_conf.h	usbh_conf.c

Simple Scheduling – Super Loop



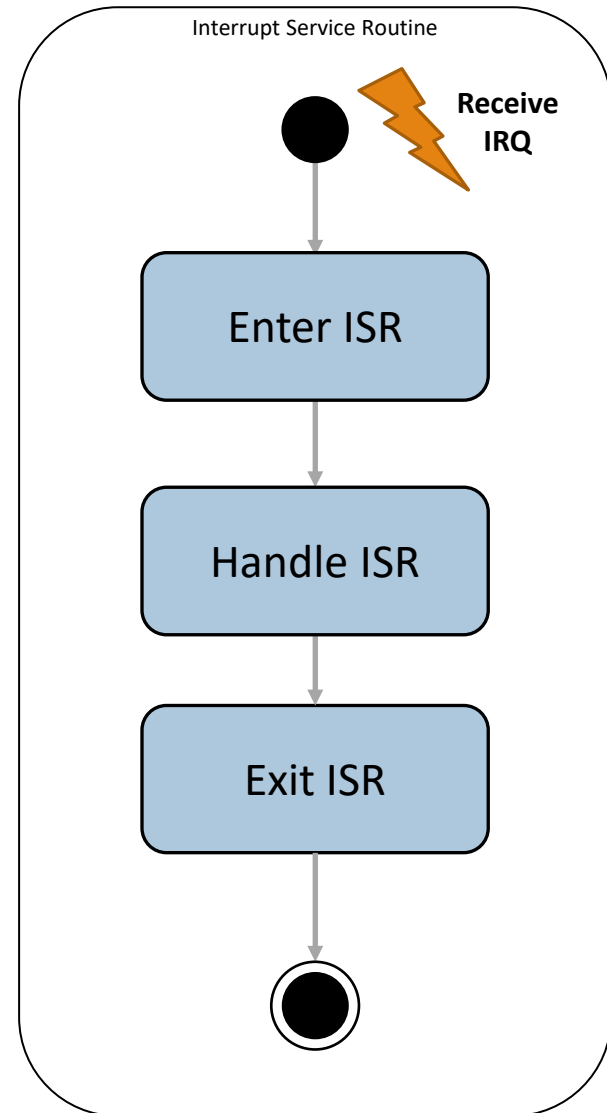
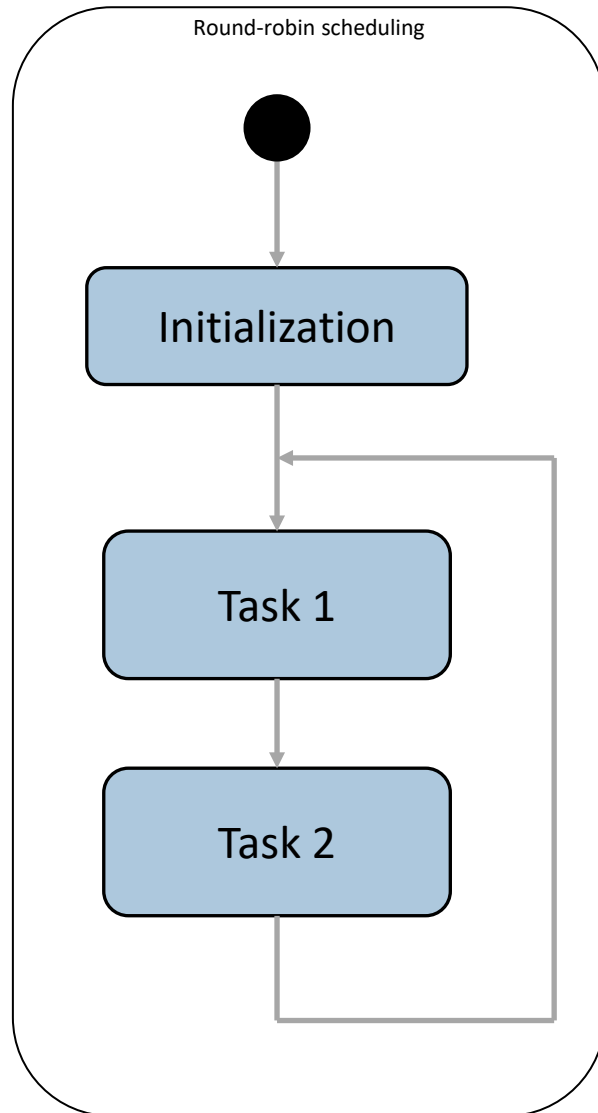
```
int main()
{
    System_Init();

    // Begin round robin scheduling
    while (1)
    {
        Task_1();

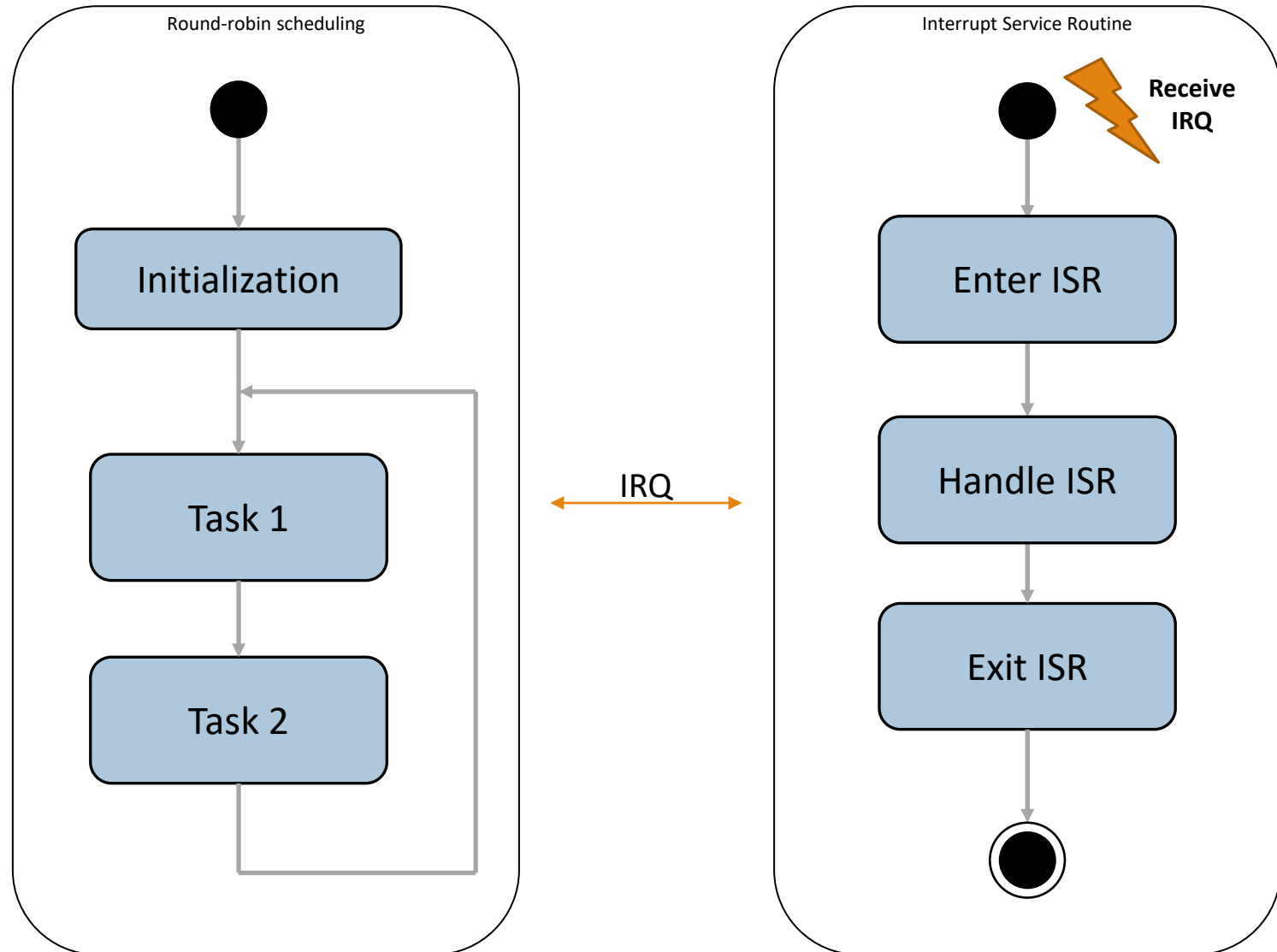
        Task_2();
    }

    return 0;
}
```

Simple Scheduling – Interrupts



Simple Scheduling – Interrupts



State Machines

“State Machines Reduce Spaghetti Code”

A State Machine captures the logic of the system in a structured way

Key concepts are

Event Some external or internal inputs which the system shall respond to.

State An encapsulation of the history of past events

The State Machine describes how the *next state* is determined by the *current state* and incoming *events*

Lion Cage Example

1. Inputs: Sensors g_1 and g_0

2. Outputs: u_1 and u_0

$u_1 = '1'$ if indoors otherwise $'0'$

$u_0 = '1'$ if outside otherwise $'0'$

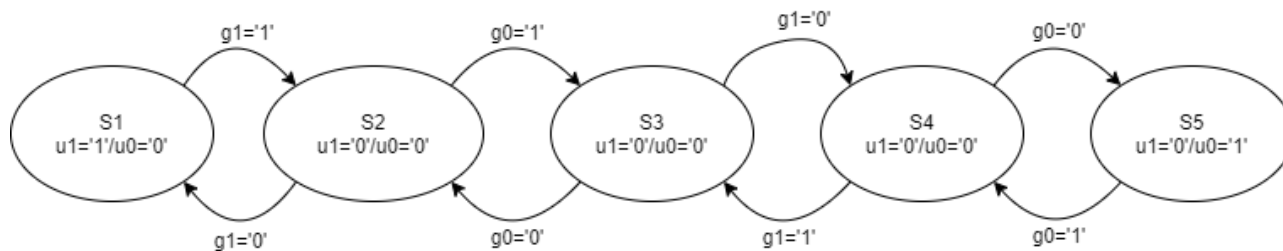
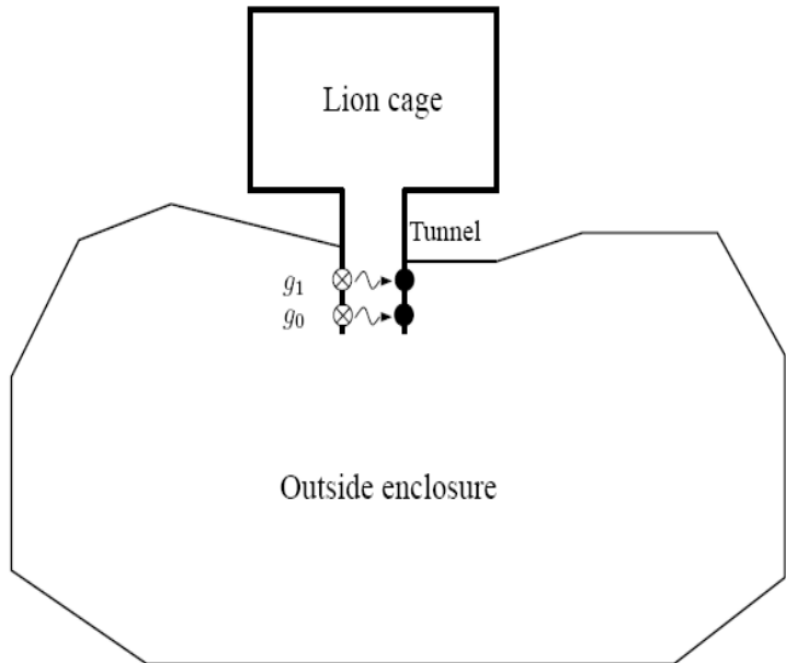
$S_1 = \text{"Lion in the cage"}$

$S_2 = \text{"Lion almost in the cage"}$

$S_3 = \text{"Lion in the tunnel"}$

$S_4 = \text{"Lion almost outdoors"}$

$S_5 = \text{"Lion outdoors"}$



State Machines – Moore vs Mealy

Moore

- A Moore machine performs actions when entering a state. Each state may have its own entry action.

Mealy

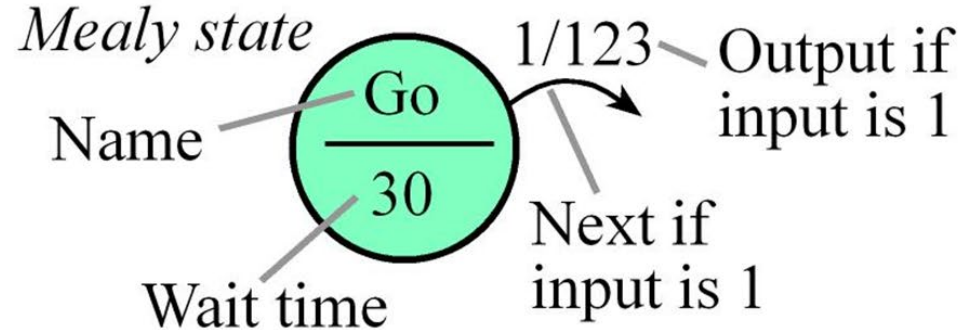
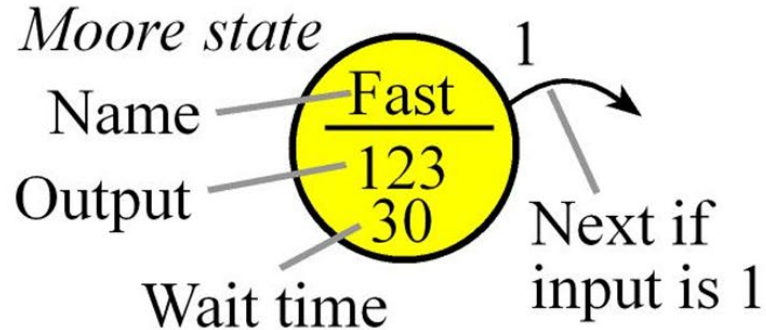
- A Mealy machine performs actions on transitions. Each transition in a state machine may invoke a unique action.

Note: Both Moore and Mealy can be used at the same time.

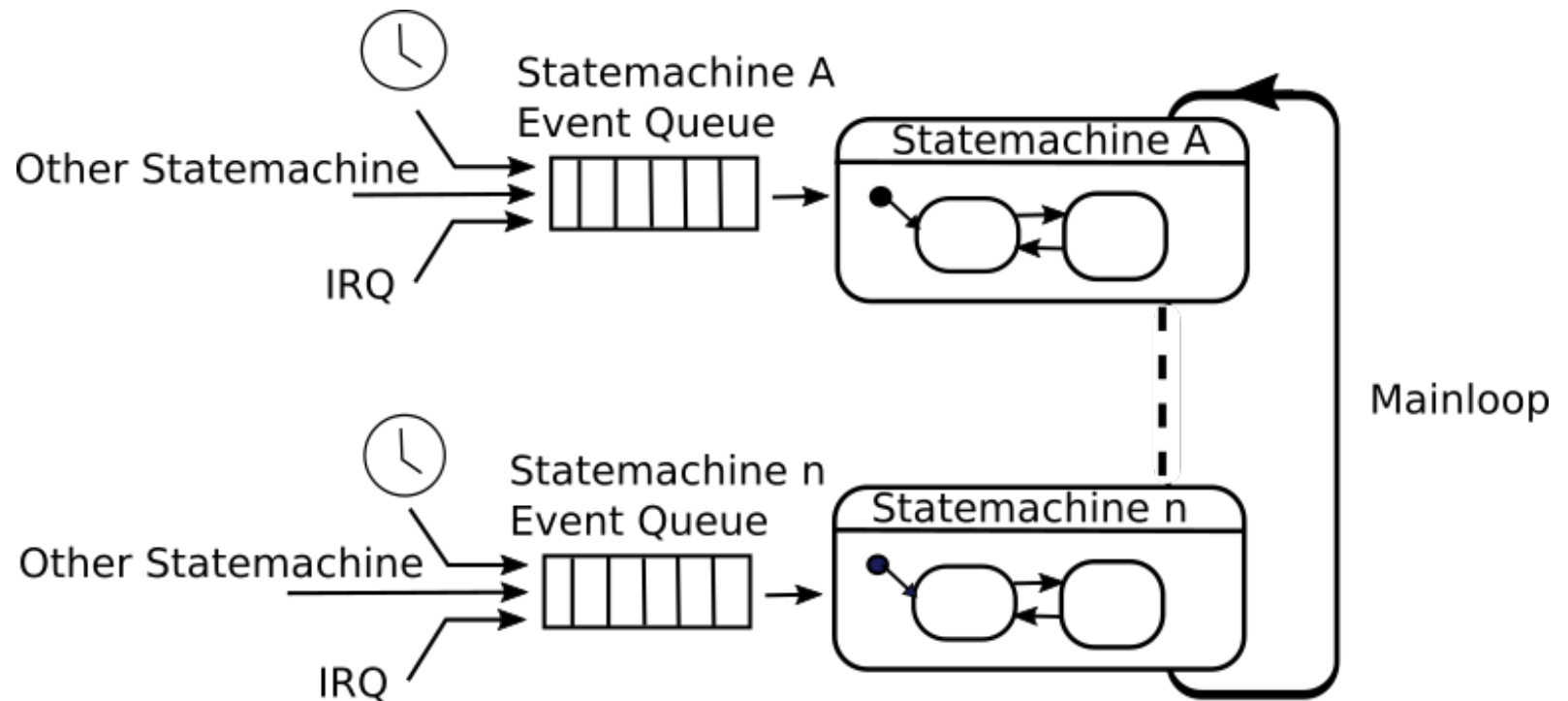
State Machines – Moore vs Mealy

Moore State Machine	Mealy State Machine
Outputs depend on current state only <i>outputs = F(currentState);</i> <i>nextState = F(inputs, currentState);</i>	Outputs depend on current state and the inputs <i>outputs = F(inputs, currentState);</i> <i>nextState = F(inputs, currentState);</i>
Generally, it has more states than Mealy	Generally, it has fewer states than Moore
Output does not react immediately to input change (one clock cycle later)	Outputs have immediate reaction to inputs
Output is placed on states	Output is placed on transitions
Easy to design	It is difficult to design

State Machines – Moore vs Mealy

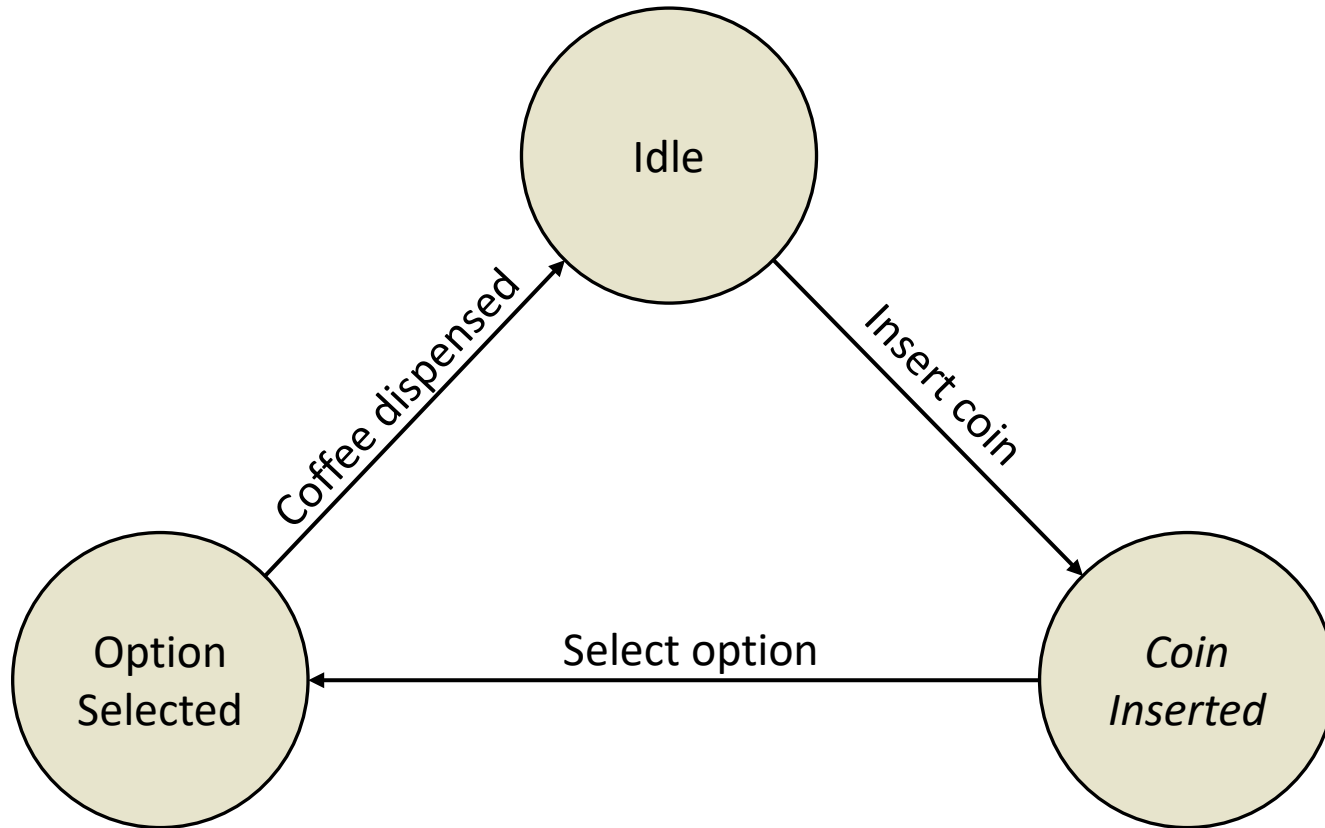


State Machines



Simple State Machine – Example

Coffee Machine



Simple State Machine – Example

```
typedef enum {  
    IDLE_STATE,  
    COIN_INSERTED_STATE,  
    OPTION_SELECTED_STATE,  
    NUM_STATES  
} state_e;
```

```
typedef enum {  
    INSERT_COIN_EVENT,  
    SELECT_OPTION_EVENT,  
    COFFEE_READY_EVENT,  
    NUM_EVENTS  
} event_e;
```

```
state_e state = IDLE_STATE;  
state_e next_state = state;  
event_e event;
```

Simple State Machine – Example

```
while (1)
{
    event = read_event();
    if (state == IDLE_STATE)
    {
        if (event == INSERT_COINT_EVENT)
        {
            next_state = insert_coin_event_handler();
        }
    }
    else if (state == COIN_INSERTED_STATE)
    {
        if (event == SELECT_OPTION_EVENT)
        {
            next_state = select_option_event_handler();
        }
    }
    else if (state == OPTION_SELECTED_STATE)
    {
        if (event == COFFEE_READY_EVENT)
        {
            next_state = coffee_ready_event_handler();
        }
    }

    state = next_state;
}
```

State Transition Matrix

STATE	Event_1	Event_2	Event_3	OUTPUT
State_1	State_2			OUT_1
State_2	State_1	State_5	State_3	OUT_2
State_3			State_4	OUT_3
State_4			State_5	OUT_4
State_5		State_2	State_2	OUT_5

Simple State Machine – Example

```
state_e (*state_table[NUM_STATES][NUM_EVENTS])(void) = {
    {insert_coin_event_handler, error_handler, error_handler},
    {error_handler, select_option_event_handler, error_handler},
    {error_handler, error_handler, coffee_ready_event_handler}
};

while (1)
{
    event = read_event();
    if (event >= 0 && (event < NUM_EVENTS))
    {
        next_state = state_table[state][event]();
        state = next_state;
    }
}
```

Example – Washer Machine



Example – Washer Machine

```
// Function declaration
```

```
// Variables
```

```
void main (void) {  
    SystemInit();  
    while(1) {  
        Wash(); // State Machine for Washing  
        Heat(); // State Machine for Heating  
    }  
}
```

Example – Washer Machine

```
typedef enum {STATE1, STATE2, STATE3} state_ex_t;
```

```
typedef enum {s_OFF, s_FILL, s_WASH, s_EMPTY} state_washer_t;
```

```
typedef enum {s_OFF_H, s_WAIT1, s_HEAT, s_WAIT2} state_heat_t;
```

```
typedef enum {NONE, ONOFF, FILLED, EMPTIED} event_t;
```

Example – Washer Machine

```
static state_washer_t washState = s_OFF;
```

```
switch (washState) {
```

```
    case s_OFF:
```

```
        Do something
```

```
        if (event == ONOFF) {
```

```
            washState = s_FILL;
```

```
        }
```

```
        break;
```

```
    case s_FILL:
```

```
        Do something
```

```
        if (event == FILLED) {
```

```
            washState = s_WASH;
```

```
        }
```

```
        break;
```

```
    case s_WASH:
```

```
        Do something
```

```
        if (washing_process == READY) {
```

```
            washState = s_EMPTY;
```

```
        }
```

```
        break;
```

```
    case s_EMPTY:
```

```
        Do something
```

```
        if (event == EMPTIED) {
```

```
            washState = s_OFF;
```

```
        }
```

```
        break;
```

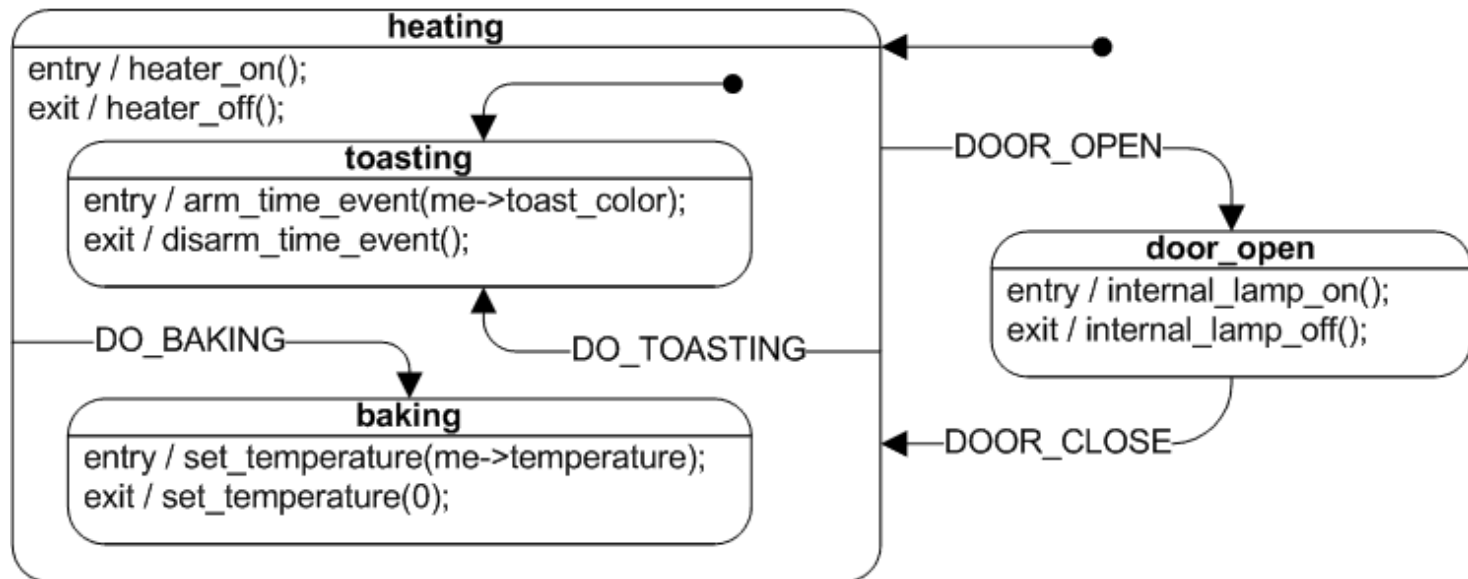
```
    default: washState = s_OFF;
```

```
        break;
```

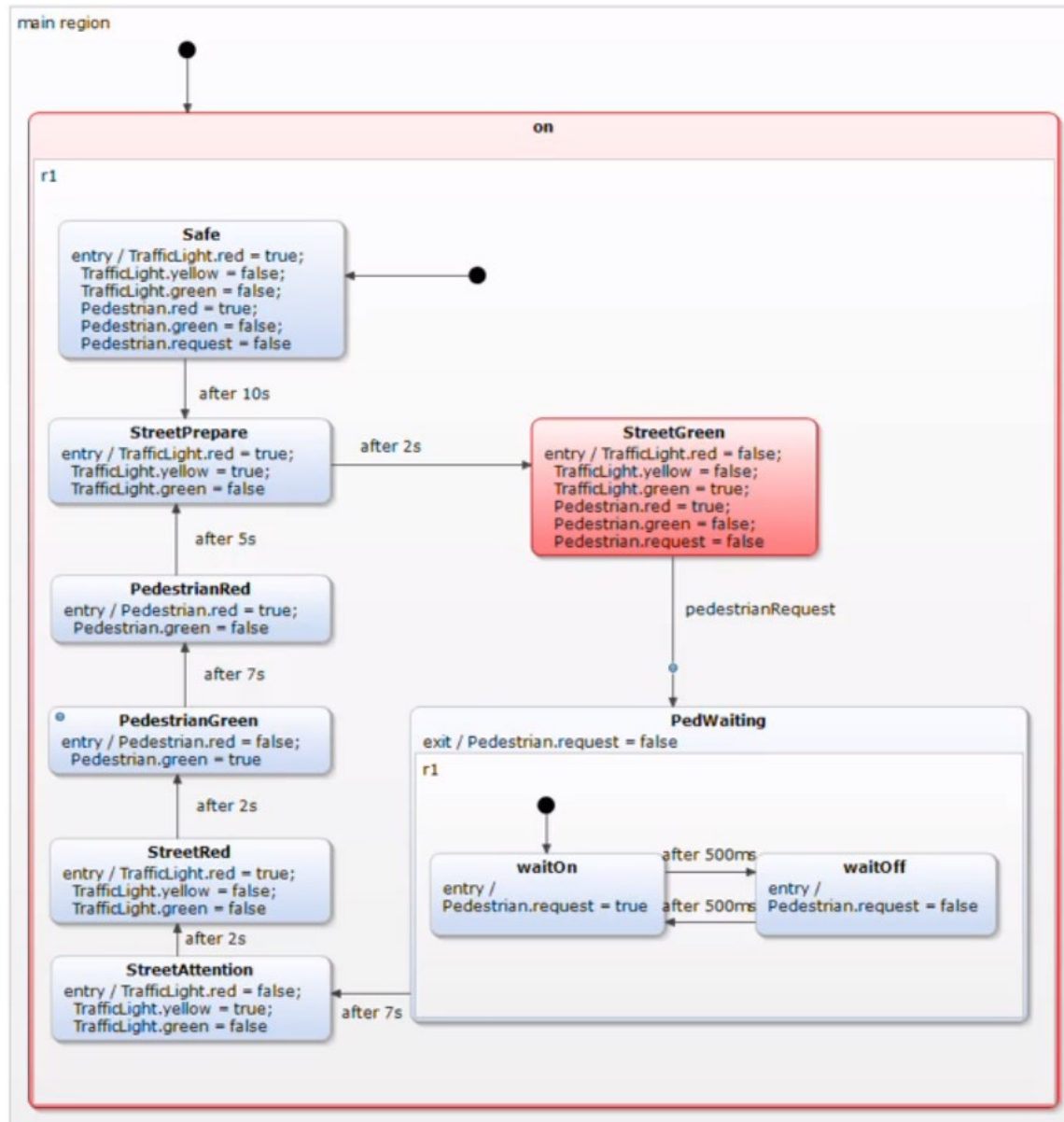
```
}
```

NEVER A WHILE INSIDE A STATE MACHINE

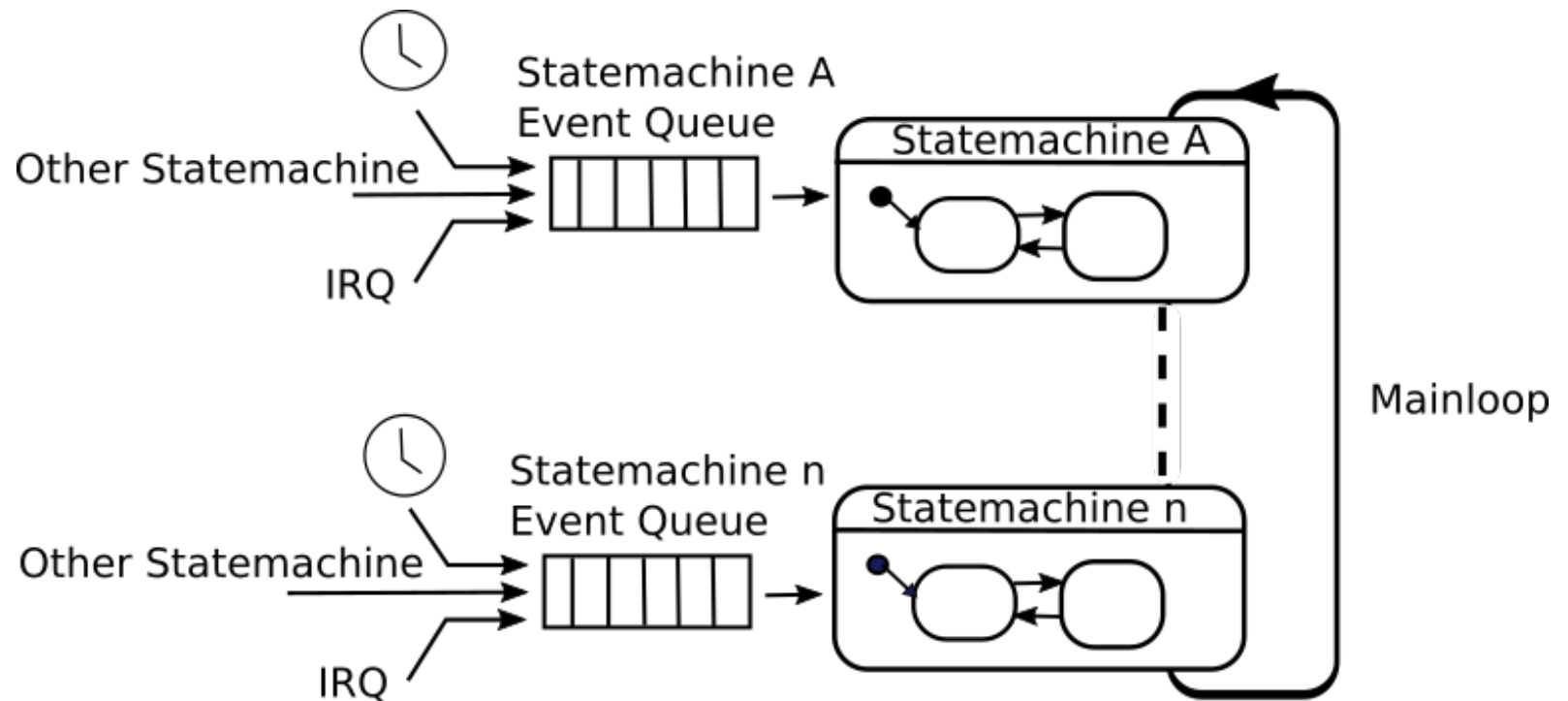
Simple State Machine – Example



State Machine – Example



State Machines – Revisited



Microcomputer Engineering

Questions?

Contact information

Andreas Axelsson

Email: andreas.axelsson@ju.se

Mobile: 0709-467760