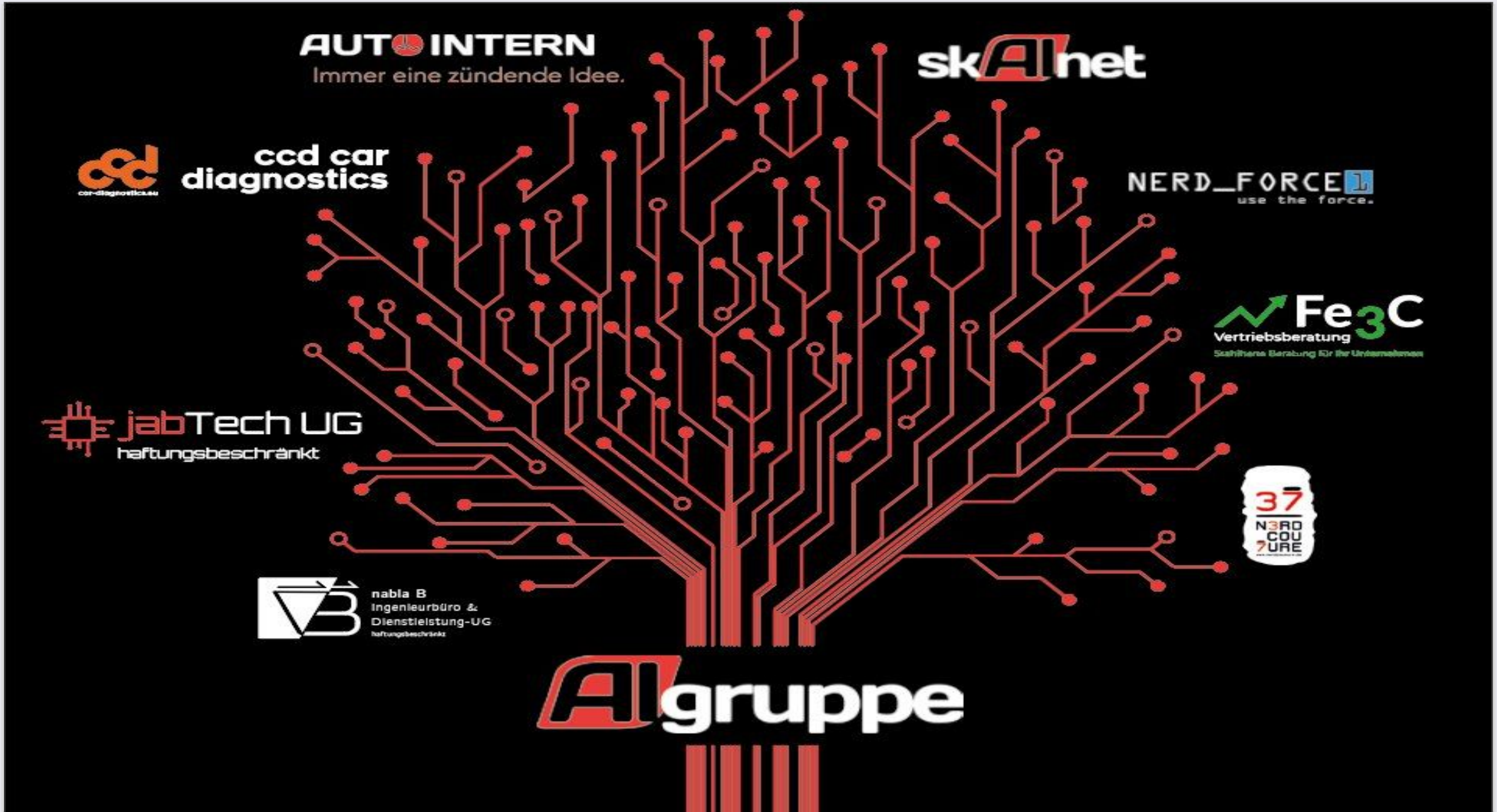
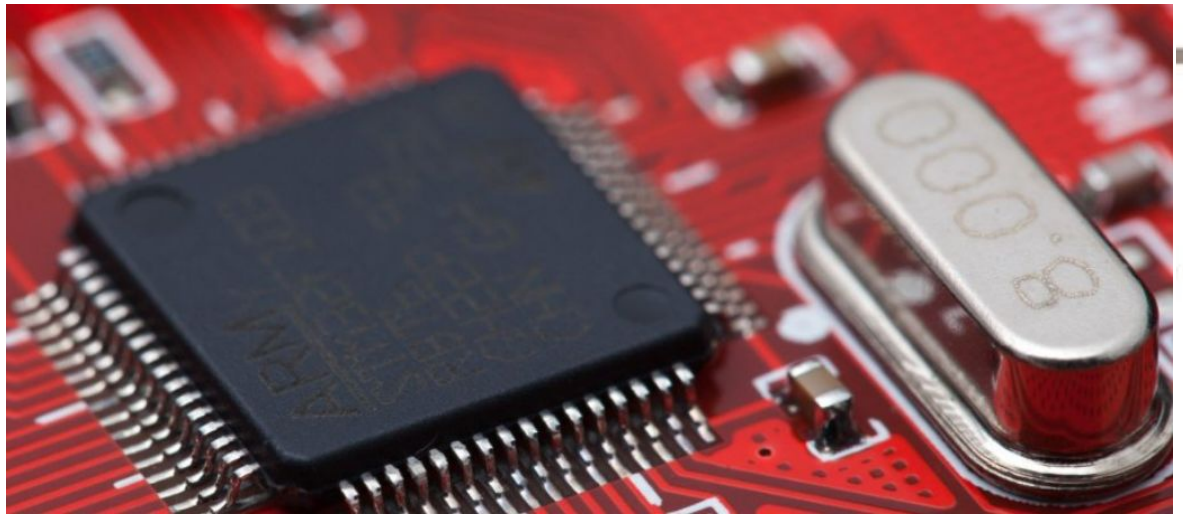
@odinthenerd

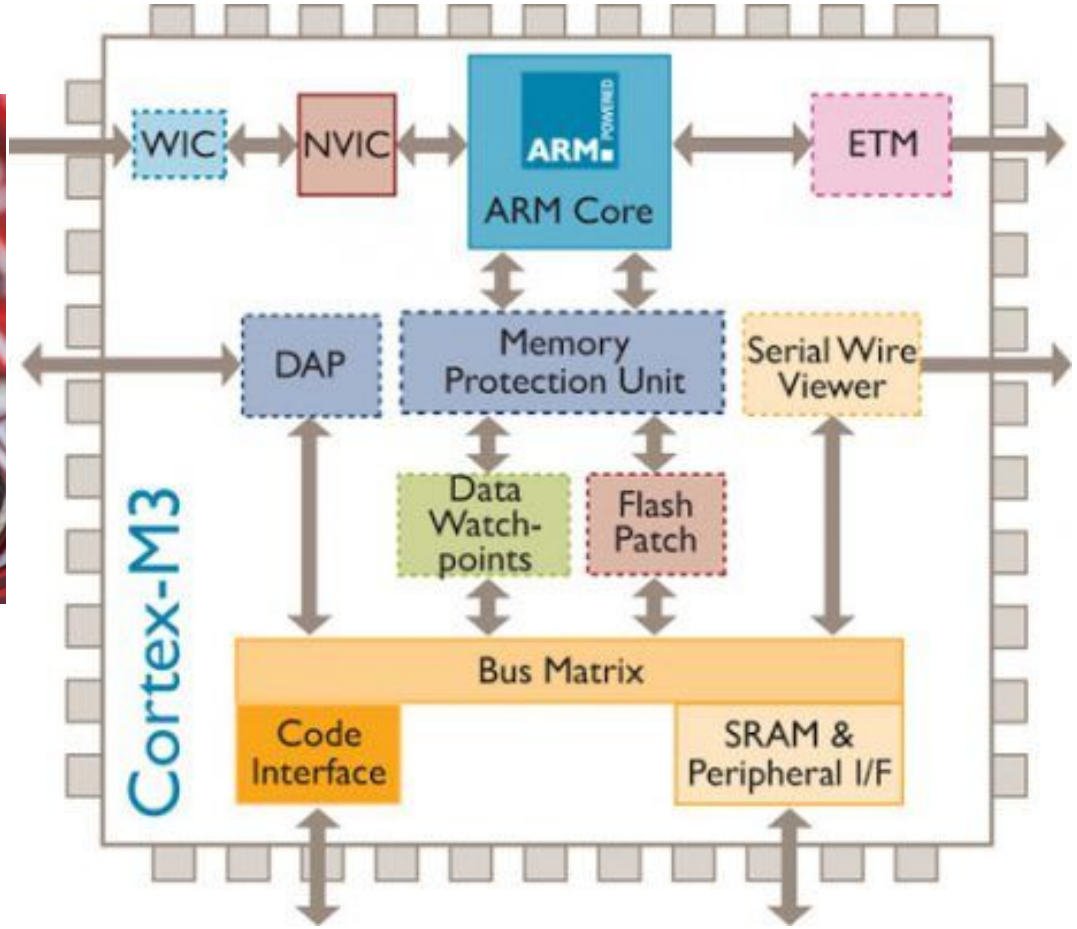
– not the god

@odinthenerd

– not the god


AUTO INTERN
Immer eine zündende Idee.
skAInet
ccd car diagnostics
car-diagnostics.eu
NERD_FORCE
use the force.
Fe3C
Vertriebsberatung
jabTech UG
haftungsbeschränkt
nabla B
Ingenieurbüro &
Dienstleistung-UG
haftungsbeschränkt
37
N3RD COU7URE
AIgruppe




WIC
NVIC
ARM
ARM Core
ETM
DAP
Memory Protection Unit
Serial Wire Viewer
Data Watch-points
Flash Patch
Cortex-M3
Bus Matrix
Code Interface
SRAM & Peripheral I/F

Race conditions

Threadding

```
int main()
{
  int bla;
  do_stuff();
  do_nothing();
  return 0;
}
```

Multicore

```
int main()
{
  int bla;
  do_stuff();
  do_nothing();
  return 0;
}
```

```
void threaddy_kruger(){
  do_bad_stuff();
  look_scary();
  do_evil_stuff();
  brush_teeth();
  do_taxes();
}
```

Multicore

```
int main()
{
  int bla;
  do_stuff();
  do_nothing();
  return 0;
}
```

```
void threaddy_kruger(){
  do_bad_stuff();
  look_scary();
  do_evil_stuff();
  brush_teeth();
  do_taxes();
}
```

Context switching

```
int main()
{
  int bla;


  do_stuff();
  do_nothing();









  return 0;
}
```

```
void threaddy_kruger(){
  do_bad_stuff();


  look_scary();
  do_evil_stuff();
  brush_teeth();
  do_taxes();
}
```

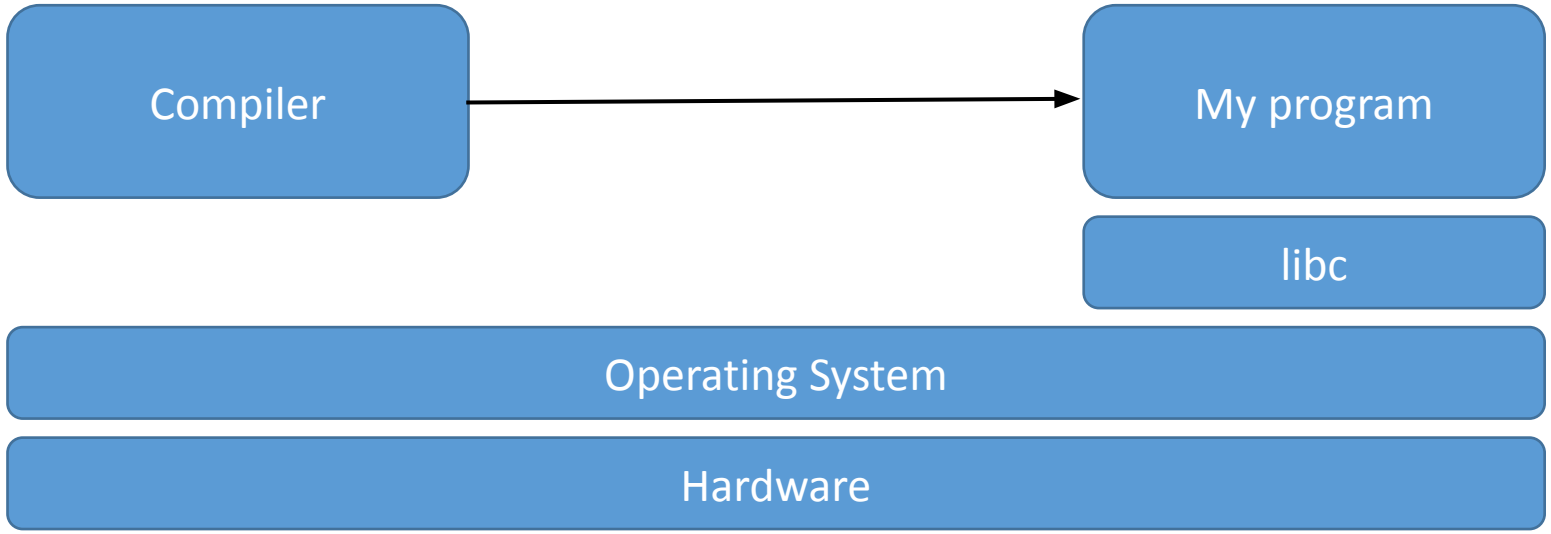
Context switching

```
int main()
{
  int bla;


  do_stuff();

  do_nothing();




  return 0;
}
```

```
void threaddy_kruger(){

  do_bad_stuff();
  look_scary();

  do_evil_stuff();
  brush_teeth();
  do_taxes();
}
```


I DO MY OWN
STUNTS

Lib c - an adopter pattern for hardware



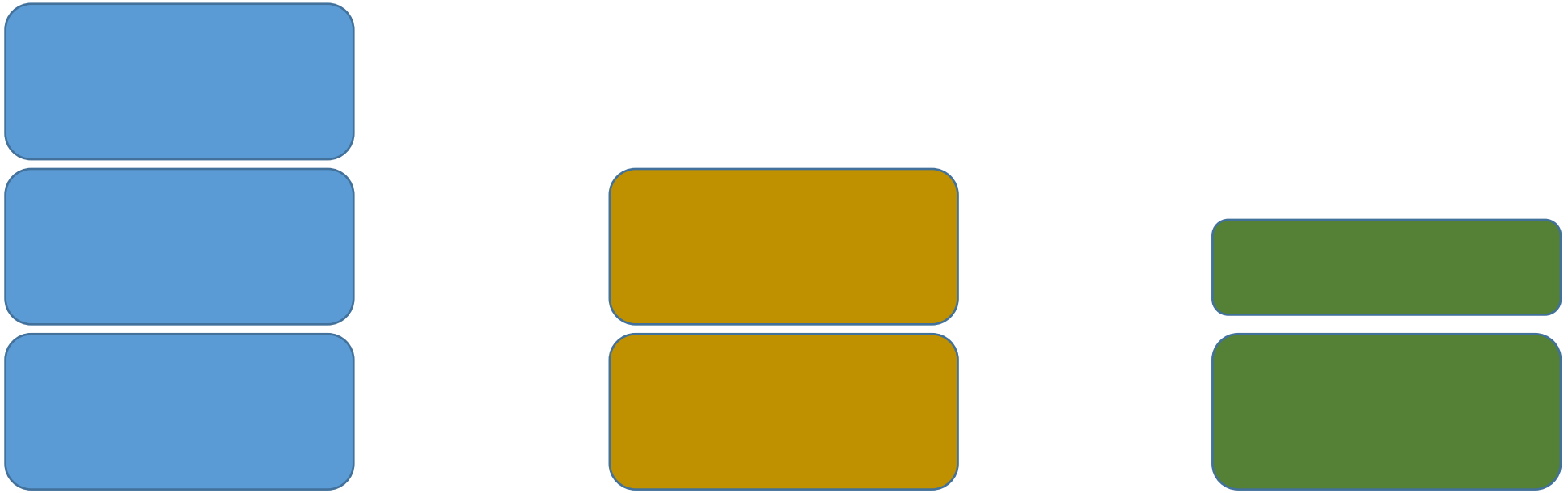
Interrupts (preemptive run to completion)

```
int main()
{
  int bla;









  do_stuff();
  do_nothing();
  return 0;
}
```

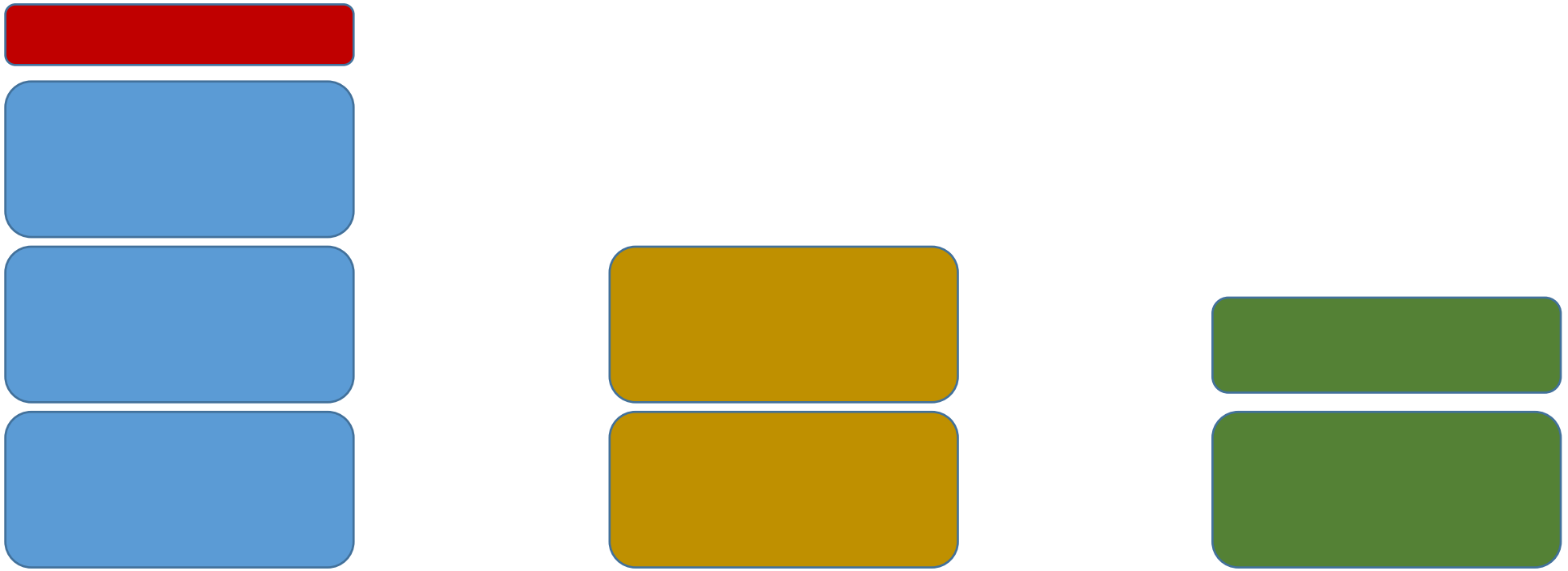
```
void threaddy_kruger(){
  do_bad_stuff();
  look_scary();
  do_evil_stuff();
  brush_teeth();
  do_taxes();
}
```

Interrupts share stack

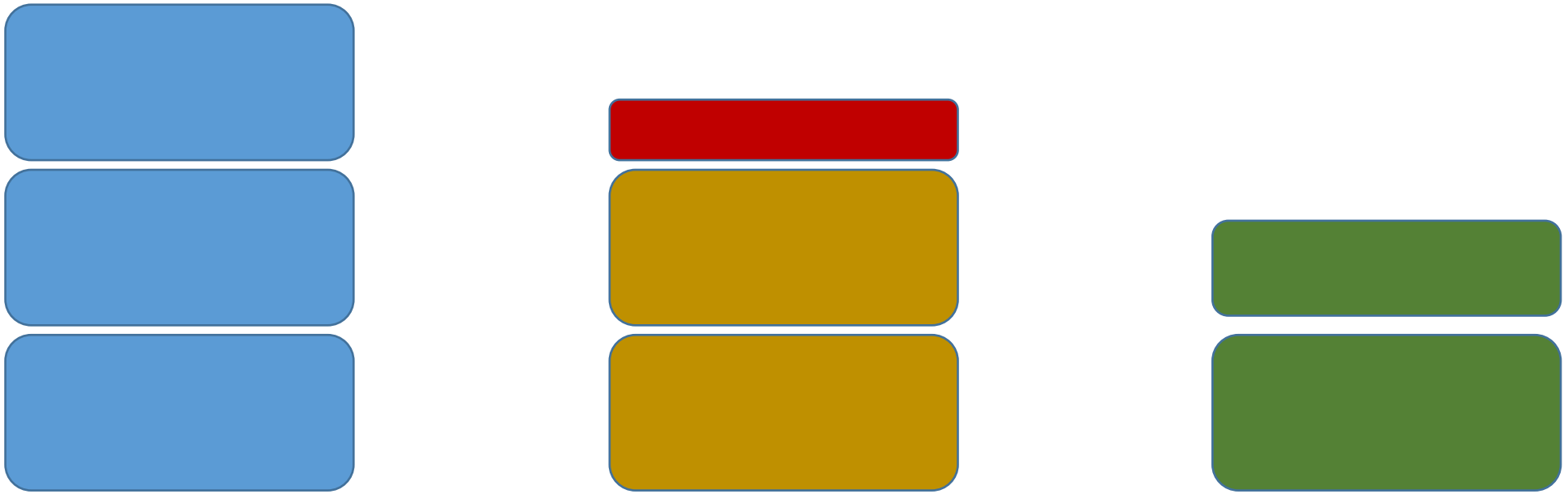
Interrupts share stack



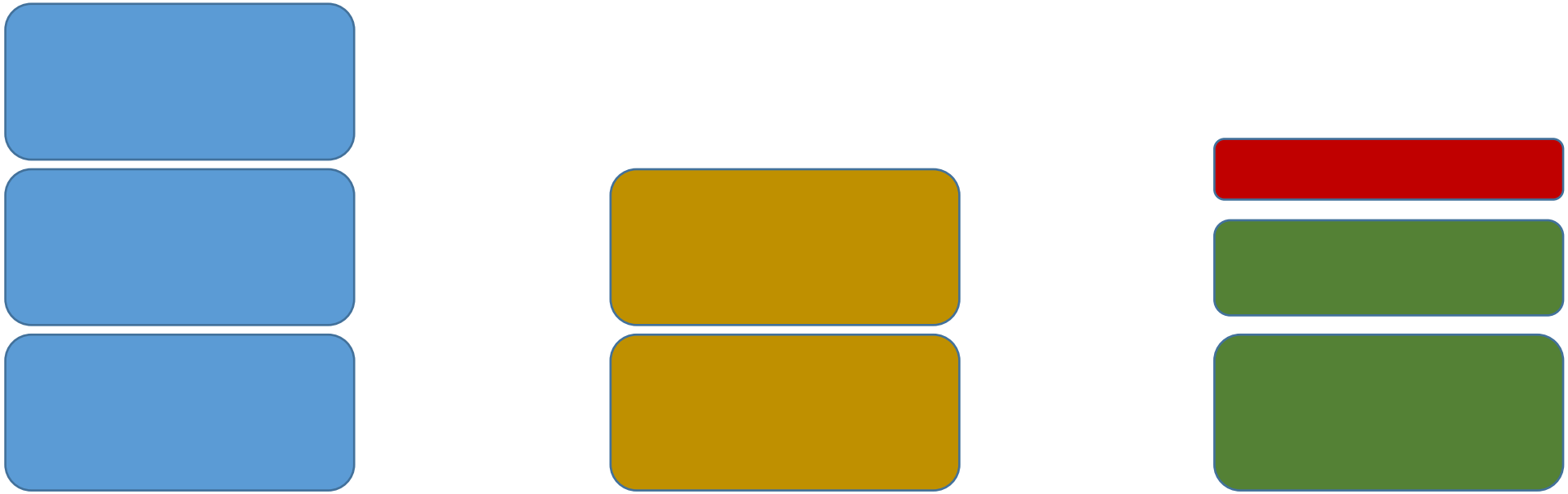
Interrupts share stack



Interrupts share stack



Interrupts share stack



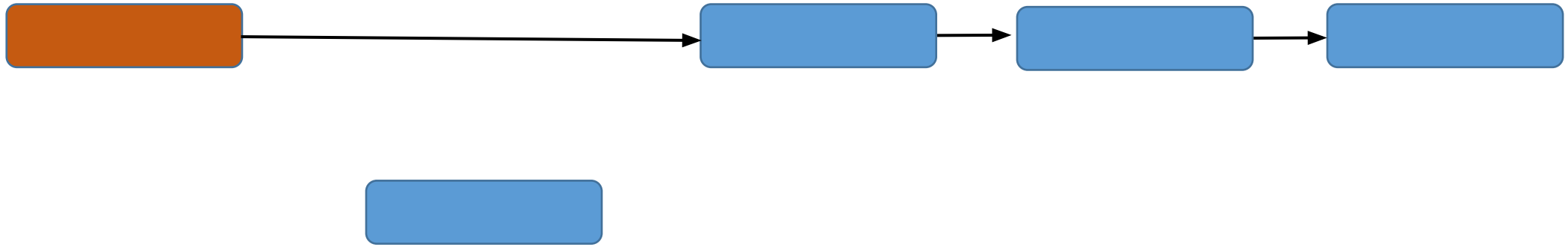
Signals share stack

ldrex strex

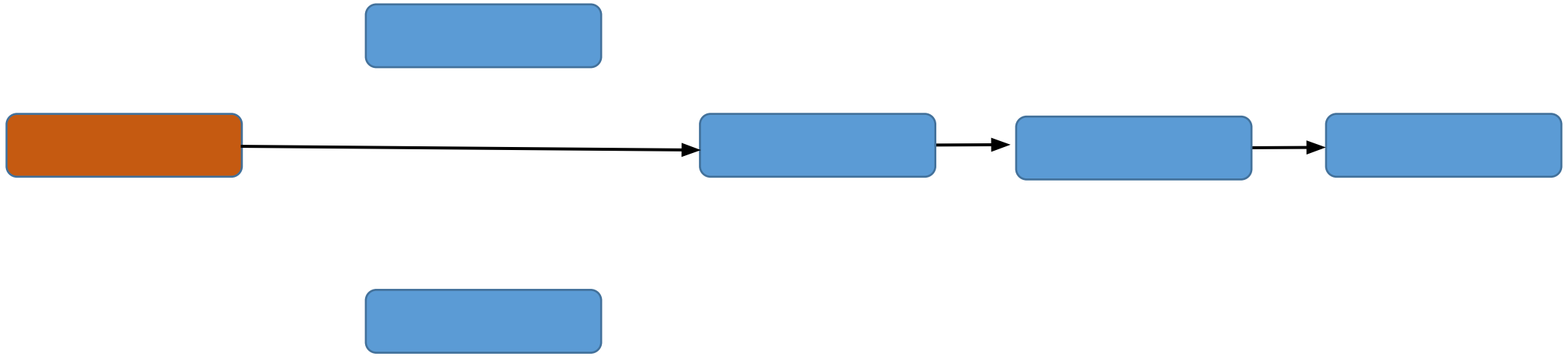
```
LDREX R1, [R0]
STREX R2, R1, [R0]
```

ldrex strex

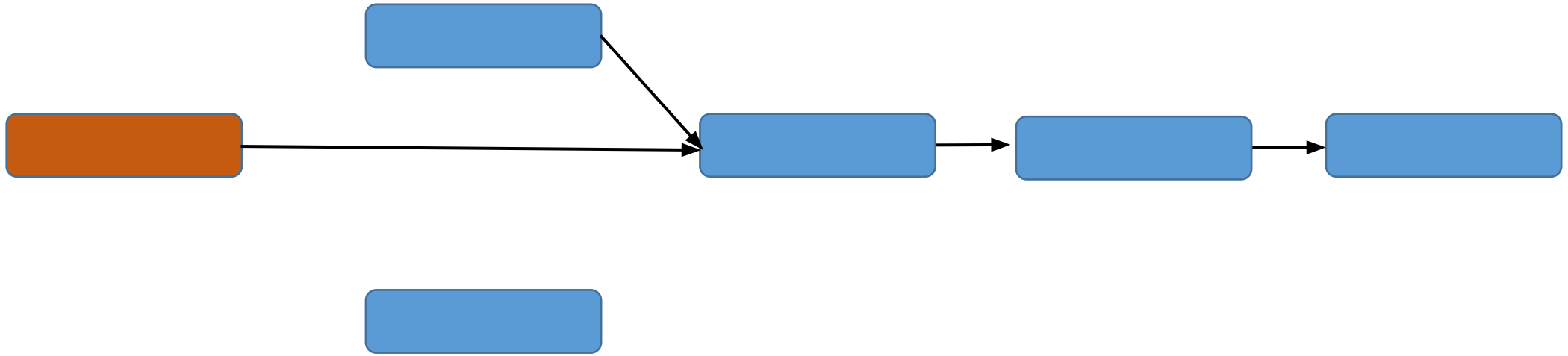
ldrex strex



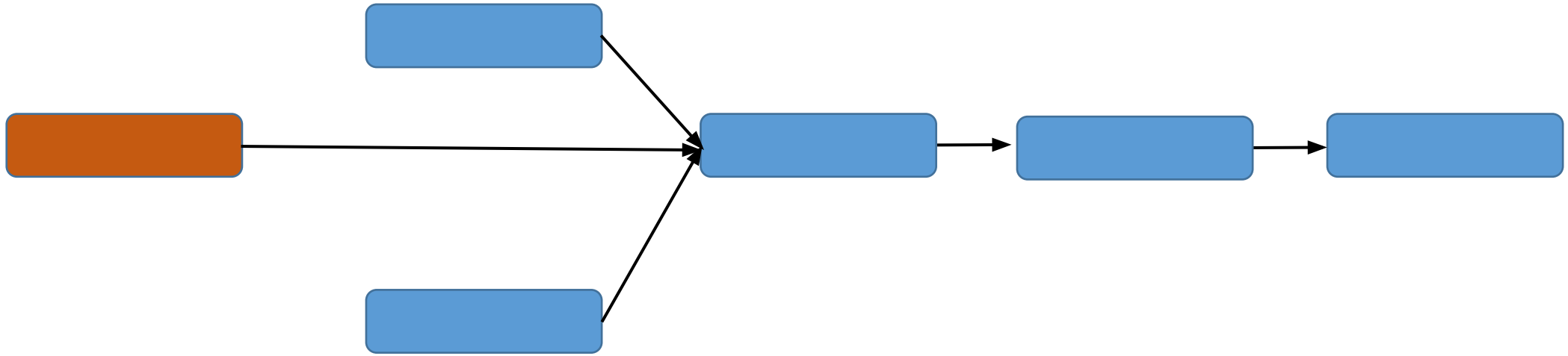
ldrex strex



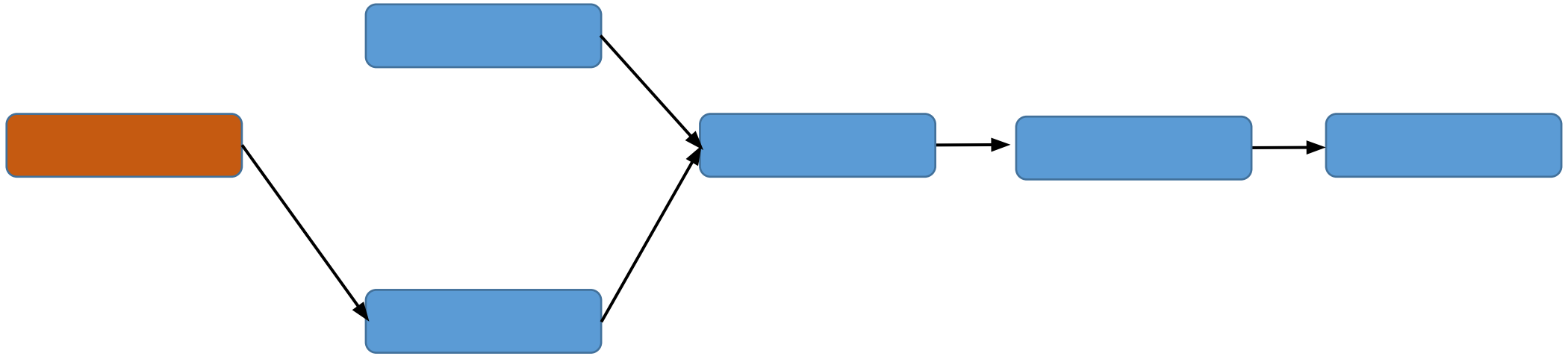
ldrex strex



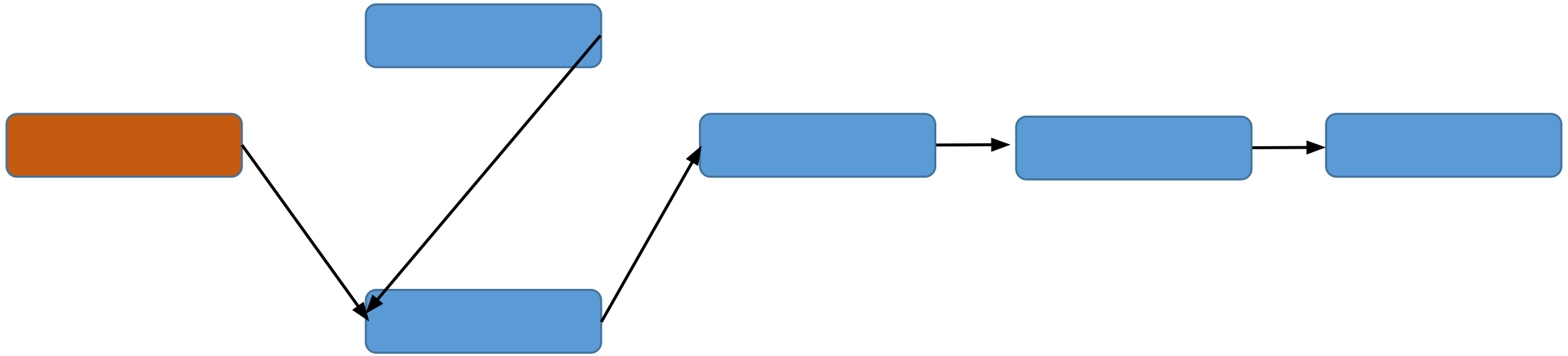
ldrex strex



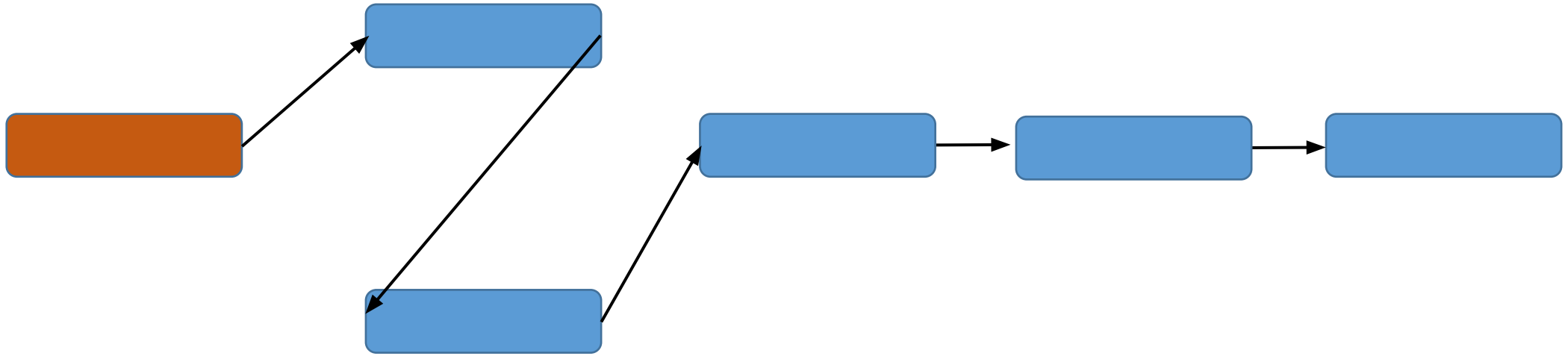
ldrex strex

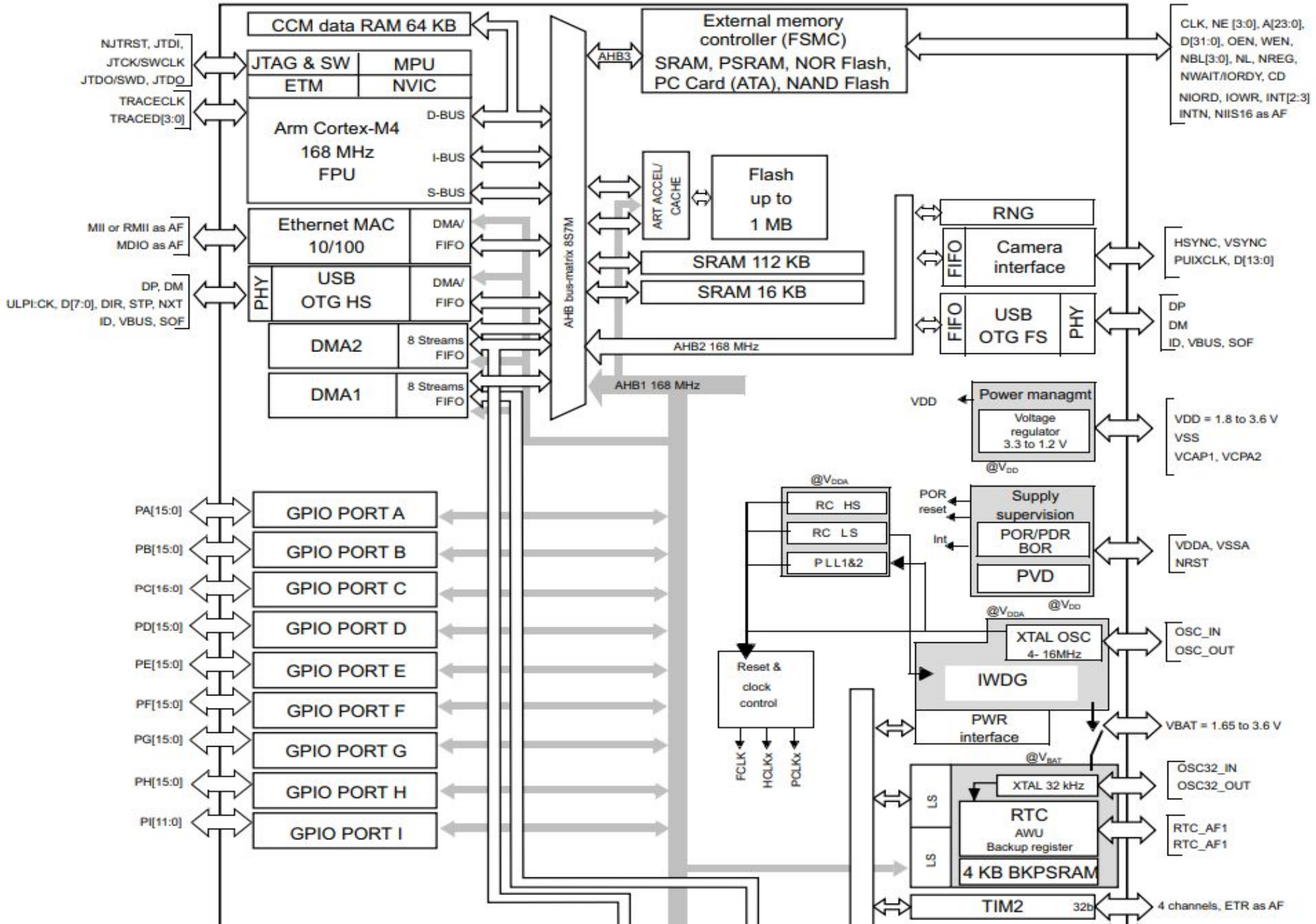


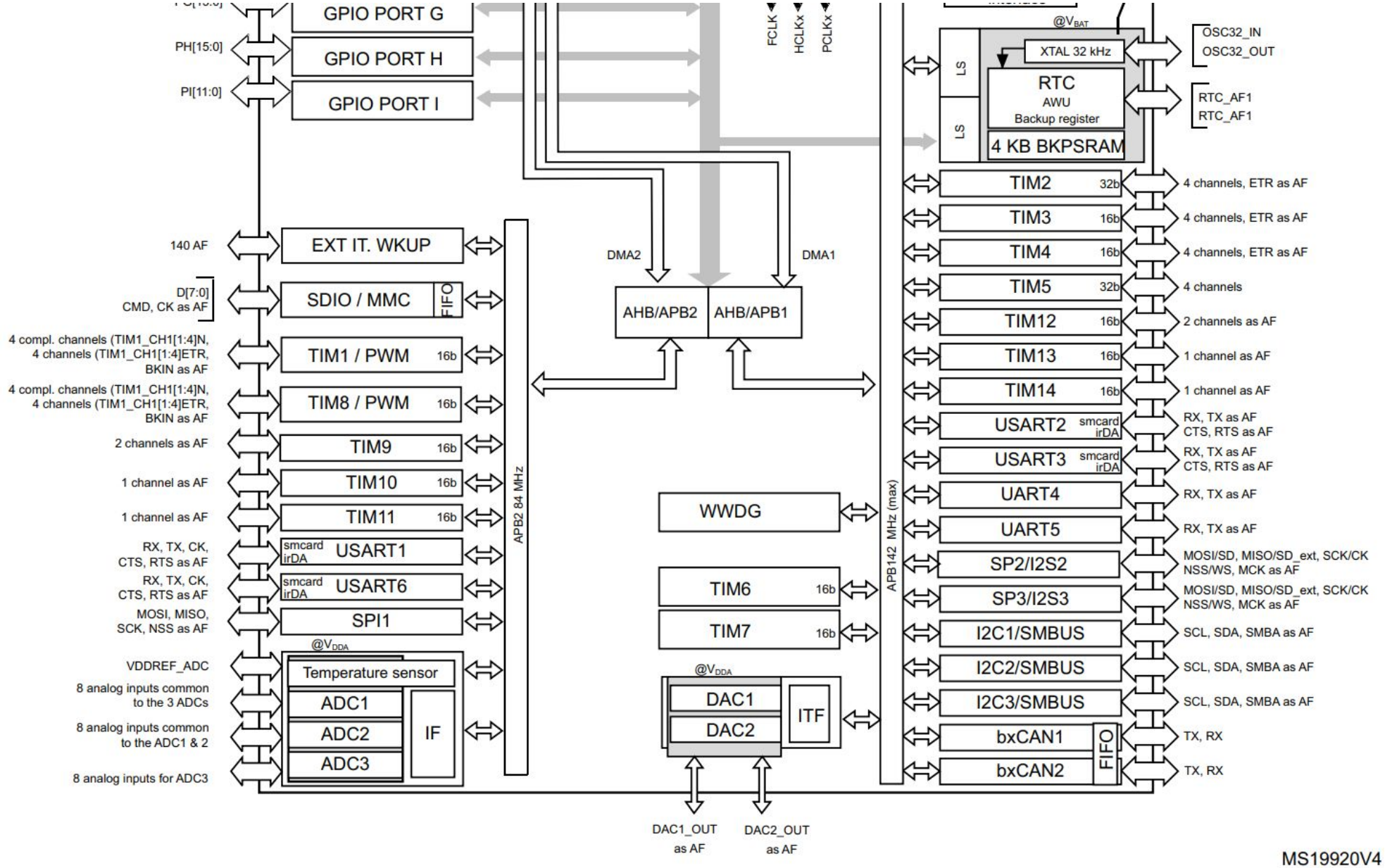
ldrex strex



ldrex strex




CCM data RAM 64 KB
NJTRST, JTDI, JTCK/SWCLK JTDO/SWD, JTDO
JTAG & SW
MPU
ETM
NVIC
TRACECLK TRACED[3:0]
Arm Cortex-M4 168 MHz FPU
D-BUS
I-BUS
S-BUS
AHB3
External memory controller (FSMC) SRAM, PSRAM, NOR Flash, PC Card (ATA), NAND Flash
CLK, NE [3:0], A[23:0], D[31:0], OEN, WEN, NBL[3:0], NL, NREG, NWAIT/IORDY, CD
NIORD, IOWR, INT[2:3] INTN, NIIS16 as AF
AHB bus-matrix 8S7M
ART ACCEL/ CACHE
Flash up to 1 MB
SRAM 112 KB
SRAM 16 KB
MII or RMII as AF MDIO as AF
Ethernet MAC 10/100
DMA/ FIFO
DP, DM ULPI:CK, D[7:0], DIR, STP, NXT ID, VBUS, SOF
PHY
USB OTG HS
DMA2
8 Streams FIFO
DMA1
AHB2 168 MHz
AHB1 168 MHz
RNG
FIFO
Camera interface
HSYNC, VSYNC PUIXCLK, D[13:0]
USB OTG FS
DP DM ID, VBUS, SOF
VDD
Power managmt
Voltage regulator 3.3 to 1.2 V
VDD = 1.8 to 3.6 V VSS VCAP1, VCPA2
PA[15:0]
GPIO PORT A
PB[15:0]
GPIO PORT B
PC[15:0]
GPIO PORT C
PD[15:0]
GPIO PORT D
PE[15:0]
GPIO PORT E
PF[15:0]
GPIO PORT F
PG[15:0]
GPIO PORT G
PH[15:0]
GPIO PORT H
PI[11:0]
GPIO PORT I
RC HS
RC LS
PLL1&2
POR reset
Int
Supply supervision
POR/PDR BOR
PVD
VDDA, VSSA NRST
Reset & clock control
FCLK
HCLKx
PCLKx
XTAL OSC 4- 16MHz
IWDG
OSC_IN OSC_OUT
PWR interface
VBAT = 1.65 to 3.6 V
XTAL 32 kHz
RTC AWU Backup register
4 KB BKPSRAM
LS
OSC32_IN OSC32_OUT
RTC_AF1 RTC_AF1
TIM2
32b
4 channels, ETR as AF


GPIO PORT G
GPIO PORT H
GPIO PORT I
PH[15:0]
PI[11:0]
FCLK
HCLKx
PCLKx
@VBAT
XTAL 32 kHz
RTC
AWU
Backup register
4 KB BKPSRAM
LS
LS
OSC32_IN
OSC32_OUT
RTC_AF1
RTC_AF1
140 AF
EXT IT. WKUP
D[7:0]
CMD, CK as AF
SDIO / MMC
FIFO
DMA2
DMA1
AHB/APB2
AHB/APB1
4 compl. channels (TIM1_CH1[1:4]N, 4 channels (TIM1_CH1[1:4]ETR, BKIN as AF
TIM1 / PWM 16b
4 compl. channels (TIM1_CH1[1:4]N, 4 channels (TIM1_CH1[1:4]ETR, BKIN as AF
TIM8 / PWM 16b
2 channels as AF
TIM9 16b
1 channel as AF
TIM10 16b
1 channel as AF
TIM11 16b
RX, TX, CK, CTS, RTS as AF
smcard irDA USART1
RX, TX, CK, CTS, RTS as AF
smcard irDA USART6
MOSI, MISO, SCK, NSS as AF
SPI1
APB2 84 MHz
@VDDA
VDDREF_ADC
8 analog inputs common to the 3 ADCs
8 analog inputs common to the ADC1 & 2
8 analog inputs for ADC3
Temperature sensor
ADC1
ADC2
ADC3
IF
WWDG
TIM6 16b
TIM7 16b
@VDDA
DAC1
DAC2
ITF
DAC1_OUT as AF
DAC2_OUT as AF
APB142 MHz (max)
TIM2 32b
TIM3 16b
TIM4 16b
TIM5 32b
TIM12 16b
TIM13 16b
TIM14 16b
USART2 smcard irDA
USART3 smcard irDA
UART4
UART5
SP2/I2S2
SP3/I2S3
I2C1/SMBUS
I2C2/SMBUS
I2C3/SMBUS
bxCAN1
bxCAN2
FIFO
4 channels, ETR as AF
4 channels, ETR as AF
4 channels, ETR as AF
4 channels
2 channels as AF
1 channel as AF
1 channel as AF
RX, TX as AF
CTS, RTS as AF
RX, TX as AF
CTS, RTS as AF
RX, TX as AF
RX, TX as AF
MOSI/SD, MISO/SD_ext, SCK/CK
NSS/WS, MCK as AF
MOSI/SD, MISO/SD_ext, SCK/CK
NSS/WS, MCK as AF
SCL, SDA, SMBA as AF
SCL, SDA, SMBA as AF
SCL, SDA, SMBA as AF
TX, RX
TX, RX
MS19920V4

17.4.1 TIM1 and TIM8 control register 1 (TIMx_CR1)

Address offset: 0x00

Reset value: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 0 **CEN**: Counter enable

0: Counter disabled

1: Counter enabled

Note: External clock, gated mode and encoder mode can work only if the CEN bit has been previously set by software. However trigger mode can set the CEN bit automatically by hardware.

@odinthenerd

- Github.com
- Twitter.com
- Gmail.com
- Blogspot.com
- LinkedIn.com
- Embo.io