



Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

Armazenamento e organização de dados em Assembly

Assembly para MSP430

Daniel Moraes - Engenharia de Computação

UnB - Universidade Federal de Brasília

Daniel Moraes da Silva

E-mail: daniel8moraess@gmail.com

Professor –Dr. Cristhian Ivan Riaño Jaimes



Assembly

O que é?

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

Assembly é uma linguagem de montagem ou de baixo nível, a qual está diretamente relacionada à arquitetura do computador (hardware). Assim, cada processador possui suas próprias particularidades na linguagem assembly. Programar em assembly é basicamente substituir os códigos de máquinas por uma linguagem mais simbólica.

```
rot:  mov.b 0x007,R6    ;Move número 7 para Registrador R6  
0011000111000110101
```



Assembly

O que é?

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

Assembly é uma linguagem de montagem ou de baixo nível, a qual está diretamente relacionada à arquitetura do computador (hardware). Assim, cada processador possui suas próprias particularidades na linguagem assembly. Programar em assembly é basicamente substituir os códigos de máquinas por uma linguagem mais simbólica.

```
rot:  mov.b 0x007,R6    ;Move número 7 para Registrador R6  
0011000111000110101
```



Porque programar em assembly??

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

- Acesso direto e maior proximidade do Hardware
- Mais eficiente, não gera códigos supérfluos
- Desenvolver rotinas mais eficazes para incorporá-las em programas de alto nível
- Programas exigem menos memória e são melhores otimizados conforme a arquitetura trabalhada



Porque programar em assembly??

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

- Acesso direto e maior proximidade do Hardware
- Mais eficiente, não gera códigos supérfluos
- Desenvolver rotinas mais eficazes para incorporá-las em programas de alto nível
- Programas exigem menos memória e são melhores otimizados conforme a arquitetura trabalhada



Porque programar em assembly??

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

- Acesso direto e maior proximidade do Hardware
- Mais eficiente, não gera códigos supérfluos
- Desenvolver rotinas mais eficazes para incorporá-las em programas de alto nível
- Programas exigem menos memória e são melhores otimizados conforme a arquitetura trabalhada



Porque programar em assembly??

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

- Acesso direto e maior proximidade do Hardware
- Mais eficiente, não gera códigos supérfluos
- Desenvolver rotinas mais eficazes para incorporá-las em programas de alto nível
- Programas exigem menos memória e são melhores otimizados conforme a arquitetura trabalhada



Porque programar em assembly??

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

- Acesso direto e maior proximidade do Hardware
- Mais eficiente, não gera códigos supérfluos
- Desenvolver rotinas mais eficazes para incorporá-las em programas de alto nível
- Programas exigem menos memória e são melhores otimizados conforme a arquitetura trabalhada



O que é preciso saber para programar em assembly??

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

- Hardware: Registradores, organização da memória, periféricos, arquitetura, etc.
- Conceitos binários
- Conhecer o conjunto de instruções do processador



O que é preciso saber para programar em assembly??

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

- Hardware: Registradores, organização da memória, periféricos, arquitetura, etc.
- Conceitos binários
- Conhecer o conjunto de instruções do processador



O que é preciso saber para programar em assembly??

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

- Hardware: Registradores, organização da memória, periféricos, arquitetura, etc.
- Conceitos binários
- Conhecer o conjunto de instruções do processador



O que é preciso saber para programar em assembly??

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

- Hardware: Registradores, organização da memória, periféricos, arquitetura, etc.
- Conceitos binários
- Conhecer o conjunto de instruções do processador



MSP430

Arquitetura

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

Os MSP430 são microcontroladores RISC de 16 bits voltados para aplicações de baixo consumo de energia. A Texas Instruments chama suas placas de desenvolvimento de LaunchPads. A placa utilizada é a MSP430-F5529 cujo foco é o desenvolvimento de aplicações com conectividade USB

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

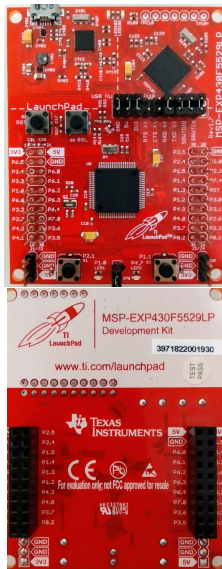


TABELA ASCII PADRÃO

	0	1	2	3	4	5	6	7
0	NUL	DLE	SP	0	@	P	.	p
1	SOH	DC1	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3	#	3	C	S	c	s
4	EOT	DC4	\$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	'	7	G	W	g	w
8	BS	CAN	(8	H	X	h	x
9	HT	EM)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[k	{
C	FF	FS	,	<	L	\	l	
D	CR	GS	-	=	M]	m	}
E	SO	RS	.	>	N	^	n	~
F	SI	UP	/	?	O	_	o	DEL

Exemplos:

símbolo 'A'	=	código ASCII 41h
símbolo 'B'	=	código ASCII 42h
símbolo 'a'	=	código ASCII 61h
símbolo 'z'	=	código ASCII 7Ah



Mostrando algumas implementações

Bubble Sort em Assembly

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

workspace_v8 - BubbleSort/main.asm - Code Composer Studio

File Edit View Navigate Project Run Scripts Window Help

Project Explorer

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

main.asm

```
23 ;-----
24 ;Main loop here
25 ;Daniel Moraes da Silva - Engenharia de Computação UnB
26 ;BubbleSort
27 ;-----
28 ;Sort:
29     mov.b   #0,R5           ;inicia R5(registror) como endereço de memória do vetor
30     mov.b   #11,R4          ;indicador de trocas>> Diz de depois de percorrer houve TROCAS
31     mov.b   #0,R6           ;tamanho do vetor + 1 >> parametro para percorrer o vetor
32     mov.b   #1,R7           ;inidice da esquerda inicia com 0
33     call    #perc           ;inidice da direita inicia com 1
34     ;subrotina para percorrer o vetor
35 perc:   dec     R4           ;decrementa 1 do tamanho do vetor >> "quando for 0 significa q o vetor foi totalmente percorrido"
36     jz      return          ;se R4 = 0 -> ir para return
37     cmp.b   vetor(R6),vetor(R7) ;compara o elemento do atual indice R7 com o atual indice de R6
38     jlo     trocar          ;se o elemento do indice R7 for menor -> ir para TROCAR/caso contrario, continua processo
39     add.b   #1,R6           ;desloca R6 uma casa para direita
40     add.b   #1,R7           ;desloca R7 uma casa para direita
41     call    #perc           ;volta a percorrer
42
43 trocar:  mov.b   vetor(R6),R8 ;mover o elemento do indice R6 para R8
44     mov.b   R8,vetor(R7),R9 ;mover o elemento do indice R7 para R9
45     mov.b   R9,vetor(R7)    ;movendo valor contido em R8 para o elemento do vetor de indice R7
46     mov.b   R9,vetor(R6)    ;movendo valor contido em R9 para o elemento do vetor de indice R6
47     add.b   #1,R6           ;desloca R6 uma casa para direita
48     add.b   #1,R7           ;desloca R7 uma casa para direita
49     add.b   #1,R10          ;adiciona 1 em R10 >> CONTADOR DE TROCA
50     call    #perc           ;volta a percorrer o vetor
51
52 ;return: mov.b   #11,R4      ;restaura o parametro usado para percorrer o vetor
53     mov.b   #0,R6           ;restaura o indice R6
54     mov.b   #1,R7           ;restaura o indice R7
55     sub.b   R6,R10          ;IDENTIFICAR SE HOUVE TROCA! subtrai o valor do contador de trocas R10
56     jz      fim            ;se for igual a zero não tem resto >> R10 = 0 = R6 logo o vetor está ordenado >> chama FIM
57     clr     R10            ;caso contrario >> limpa o indicador de trocas
58     call    #perc           ;chama subrotina perc (percorrer) percorrer novamente
59
60 fim:     jmp     $           ;trava execução
61
62 vetor:   .data             ;declaração de elementos contidos no vetor
63     .byte  3,0,1,8,7,2,5,4,6,9
64
65 ;-----
66 ; Stack Pointer definition
67 ;-----
68     .global __STACK_END
69     .sect   .stack
70
71 ;-----
72 ; Interrupt Vectors
73 ;-----
74     .sect   ".reset"       ; MSP430 RESET Vector
75     .short  RESET
```




Mostrando algumas implementações

Bubble Sort em Assembly

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

```
workspace_v8 - BubbleSort/main.asm - Code Composer Studio
File Edit View Navigate Project Run Scripts Window Help

Project Explorer
> anvore_OF
> anvore_OS
> anvore_PO
> basico_treino
> basico2_config com pull-down
> blinks
> BubbleSort (Active - Debug)
> codigao_nula
> Compara
> compara_treino
> contador0-10
> daniel_exemplo_sensor
> danieltest2
> debounce_pwm
> Ensaio 1
> Ensaio 3
> Ensaio 4
> Ex1_m1
> Ex2_m1
> Exemp_Sensor_AlunoX
> Fibonacci
> InterruptExT1
> Igual pelo botao_treino
> M2_ex_1
> M2_ex_2
> M2_ex_3
> M2_ex_4
> M2_ex_5
> M2_ex_6
> M2_ex_8
> Ordenar_Vetor_letrs
> Ordenar_Vetor2.0
> OrdenV_Errado_OK
> OrdenVe_Errado_OK
> Problema 1.0 M1
> prova
> Pv1.1
> Pv1Q3
> PWM_dut_cycle
> Q10
> Q11
> Q12_A
> Q12_B
> Q5

main.asm
1;
2; MSP430 Assembler Code Template for use with TI Code Composer Studio
3;
4;
5;
6;
7; .....
8; .....
9; .....
10; .....
11; .....
12; .....
13; .....
14; .....
15; .....
16; .....
17; .....
18; .....
19; .....
20; .....
21; .....
22; .....
23; .....
24; .....
25; .....
26; .....
27; .....
28; .....
29; .....
30; .....
31; .....
32; .....
33; .....
34; .....
35; .....
36; .....
37; .....
38; .....
39; .....
40; .....
41; .....
42; .....
43; .....
44; .....
45; .....
46; .....
47; .....
48; .....
49; .....
50; .....
51; .....
52; .....
53; .....
54; .....
55; .....
56; .....
57; .....
58; .....
59; .....
60; .....
61; .....
62; .....
63; .....
64; .....
65; .....
66; .....
67; .....
68; .....
69; .....
70; .....
71; .....
72; .....
73; .....
74; .....
75; .....
76; .....
77; .....
78; .....
79; .....
80; .....
81; .....
82; .....
83; .....
84; .....
85; .....
86; .....
87; .....
88; .....
89; .....
90; .....
91; .....
92; .....
93; .....
94; .....
95; .....
96; .....
97; .....
98; .....
99; .....
100; .....
101; .....
102; .....
103; .....
104; .....
105; .....
106; .....
107; .....
108; .....
109; .....
110; .....
111; .....
112; .....
113; .....
114; .....
115; .....
116; .....
117; .....
118; .....
119; .....
120; .....
121; .....
122; .....
123; .....
124; .....
125; .....
126; .....
127; .....
128; .....
129; .....
130; .....
131; .....
132; .....
133; .....
134; .....
135; .....
136; .....
137; .....
138; .....
139; .....
140; .....
141; .....
142; .....
143; .....
144; .....
145; .....
146; .....
147; .....
148; .....
149; .....
150; .....
151; .....
152; .....
153; .....
154; .....
155; .....
156; .....
157; .....
158; .....
159; .....
160; .....
161; .....
162; .....
163; .....
164; .....
165; .....
166; .....
167; .....
168; .....
169; .....
170; .....
171; .....
172; .....
173; .....
174; .....
175; .....
176; .....
177; .....
178; .....
179; .....
180; .....
181; .....
182; .....
183; .....
184; .....
185; .....
186; .....
187; .....
188; .....
189; .....
190; .....
191; .....
192; .....
193; .....
194; .....
195; .....
196; .....
197; .....
198; .....
199; .....
200; .....
201; .....
202; .....
203; .....
204; .....
205; .....
206; .....
207; .....
208; .....
209; .....
210; .....
211; .....
212; .....
213; .....
214; .....
215; .....
216; .....
217; .....
218; .....
219; .....
220; .....
221; .....
222; .....
223; .....
224; .....
225; .....
226; .....
227; .....
228; .....
229; .....
230; .....
231; .....
232; .....
233; .....
234; .....
235; .....
236; .....
237; .....
238; .....
239; .....
240; .....
241; .....
242; .....
243; .....
244; .....
245; .....
246; .....
247; .....
248; .....
249; .....
250; .....
251; .....
252; .....
253; .....
254; .....
255; .....
256; .....
257; .....
258; .....
259; .....
260; .....
261; .....
262; .....
263; .....
264; .....
265; .....
266; .....
267; .....
268; .....
269; .....
270; .....
271; .....
272; .....
273; .....
274; .....
275; .....
276; .....
277; .....
278; .....
279; .....
280; .....
281; .....
282; .....
283; .....
284; .....
285; .....
286; .....
287; .....
288; .....
289; .....
290; .....
291; .....
292; .....
293; .....
294; .....
295; .....
296; .....
297; .....
298; .....
299; .....
300; .....
301; .....
302; .....
303; .....
304; .....
305; .....
306; .....
307; .....
308; .....
309; .....
310; .....
311; .....
312; .....
313; .....
314; .....
315; .....
316; .....
317; .....
318; .....
319; .....
320; .....
321; .....
322; .....
323; .....
324; .....
325; .....
326; .....
327; .....
328; .....
329; .....
330; .....
331; .....
332; .....
333; .....
334; .....
335; .....
336; .....
337; .....
338; .....
339; .....
340; .....
341; .....
342; .....
343; .....
344; .....
345; .....
346; .....
347; .....
348; .....
349; .....
350; .....
351; .....
352; .....
353; .....
354; .....
355; .....
356; .....
357; .....
358; .....
359; .....
360; .....
361; .....
362; .....
363; .....
364; .....
365; .....
366; .....
367; .....
368; .....
369; .....
370; .....
371; .....
372; .....
373; .....
374; .....
375; .....
376; .....
377; .....
378; .....
379; .....
380; .....
381; .....
382; .....
383; .....
384; .....
385; .....
386; .....
387; .....
388; .....
389; .....
390; .....
391; .....
392; .....
393; .....
394; .....
395; .....
396; .....
397; .....
398; .....
399; .....
400; .....
401; .....
402; .....
403; .....
404; .....
405; .....
406; .....
407; .....
408; .....
409; .....
410; .....
411; .....
412; .....
413; .....
414; .....
415; .....
416; .....
417; .....
418; .....
419; .....
420; .....
421; .....
422; .....
423; .....
424; .....
425; .....
426; .....
427; .....
428; .....
429; .....
430; .....
431; .....
432; .....
433; .....
434; .....
435; .....
436; .....
437; .....
438; .....
439; .....
440; .....
441; .....
442; .....
443; .....
444; .....
445; .....
446; .....
447; .....
448; .....
449; .....
450; .....
451; .....
452; .....
453; .....
454; .....
455; .....
456; .....
457; .....
458; .....
459; .....
460; .....
461; .....
462; .....
463; .....
464; .....
465; .....
466; .....
467; .....
468; .....
469; .....
470; .....
471; .....
472; .....
473; .....
474; .....
475; .....
476; .....
477; .....
478; .....
479; .....
480; .....
481; .....
482; .....
483; .....
484; .....
485; .....
486; .....
487; .....
488; .....
489; .....
490; .....
491; .....
492; .....
493; .....
494; .....
495; .....
496; .....
497; .....
498; .....
499; .....
500; .....
501; .....
502; .....
503; .....
504; .....
505; .....
506; .....
507; .....
508; .....
509; .....
510; .....
511; .....
512; .....
513; .....
514; .....
515; .....
516; .....
517; .....
518; .....
519; .....
520; .....
521; .....
522; .....
523; .....
524; .....
525; .....
526; .....
527; .....
528; .....
529; .....
530; .....
531; .....
532; .....
533; .....
534; .....
535; .....
536; .....
537; .....
538; .....
539; .....
540; .....
541; .....
542; .....
543; .....
544; .....
545; .....
546; .....
547; .....
548; .....
549; .....
550; .....
551; .....
552; .....
553; .....
554; .....
555; .....
556; .....
557; .....
558; .....
559; .....
560; .....
561; .....
562; .....
563; .....
564; .....
565; .....
566; .....
567; .....
568; .....
569; .....
570; .....
571; .....
572; .....
573; .....
574; .....
575; .....
576; .....
577; .....
578; .....
579; .....
580; .....
581; .....
582; .....
583; .....
584; .....
585; .....
586; .....
587; .....
588; .....
589; .....
590; .....
591; .....
592; .....
593; .....
594; .....
595; .....
596; .....
597; .....
598; .....
599; .....
600; .....
601; .....
602; .....
603; .....
604; .....
605; .....
606; .....
607; .....
608; .....
609; .....
610; .....
611; .....
612; .....
613; .....
614; .....
615; .....
616; .....
617; .....
618; .....
619; .....
620; .....
621; .....
622; .....
623; .....
624; .....
625; .....
626; .....
627; .....
628; .....
629; .....
630; .....
631; .....
632; .....
633; .....
634; .....
635; .....
636; .....
637; .....
638; .....
639; .....
640; .....
641; .....
642; .....
643; .....
644; .....
645; .....
646; .....
647; .....
648; .....
649; .....
650; .....
651; .....
652; .....
653; .....
654; .....
655; .....
656; .....
657; .....
658; .....
659; .....
660; .....
661; .....
662; .....
663; .....
664; .....
665; .....
666; .....
667; .....
668; .....
669; .....
670; .....
671; .....
672; .....
673; .....
674; .....
675; .....
676; .....
677; .....
678; .....
679; .....
680; .....
681; .....
682; .....
683; .....
684; .....
685; .....
686; .....
687; .....
688; .....
689; .....
690; .....
691; .....
692; .....
693; .....
694; .....
695; .....
696; .....
697; .....
698; .....
699; .....
700; .....
701; .....
702; .....
703; .....
704; .....
705; .....
706; .....
707; .....
708; .....
709; .....
710; .....
711; .....
712; .....
713; .....
714; .....
715; .....
716; .....
717; .....
718; .....
719; .....
720; .....
721; .....
722; .....
723; .....
724; .....
725; .....
726; .....
727; .....
728; .....
729; .....
730; .....
731; .....
732; .....
733; .....
734; .....
735; .....
736; .....
737; .....
738; .....
739; .....
740; .....
741; .....
742; .....
743; .....
744; .....
745; .....
746; .....
747; .....
748; .....
749; .....
750; .....
751; .....
752; .....
753; .....
754; .....
755; .....
756; .....
757; .....
758; .....
759; .....
760; .....
761; .....
762; .....
763; .....
764; .....
765; .....
766; .....
767; .....
768; .....
769; .....
770; .....
771; .....
772; .....
773; .....
774; .....
775; .....
776; .....
777; .....
778; .....
779; .....
780; .....
781; .....
782; .....
783; .....
784; .....
785; .....
786; .....
787; .....
788; .....
789; .....
790; .....
791; .....
792; .....
793; .....
794; .....
795; .....
796; .....
797; .....
798; .....
799; .....
800; .....
801; .....
802; .....
803; .....
804; .....
805; .....
806; .....
807; .....
808; .....
809; .....
810; .....
811; .....
812; .....
813; .....
814; .....
815; .....
816; .....
817; .....
818; .....
819; .....
820; .....
821; .....
822; .....
823; .....
824; .....
825; .....
826; .....
827; .....
828; .....
829; .....
830; .....
831; .....
832; .....
833; .....
834; .....
835; .....
836; .....
837; .....
838; .....
839; .....
840; .....
841; .....
842; .....
843; .....
844; .....
845; .....
846; .....
847; .....
848; .....
849; .....
850; .....
851; .....
852; .....
853; .....
854; .....
855; .....
856; .....
857; .....
858; .....
859; .....
860; .....
861; .....
862; .....
863; .....
864; .....
865; .....
866; .....
867; .....
868; .....
869; .....
870; .....
871; .....
872; .....
873; .....
874; .....
875; .....
876; .....
877; .....
878; .....
879; .....
880; .....
881; .....
882; .....
883; .....
884; .....
885; .....
886; .....
887; .....
888; .....
889; .....
890; .....
891; .....
892; .....
893; .....
894; .....
895; .....
896; .....
897; .....
898; .....
899; .....
900; .....
901; .....
902; .....
903; .....
904; .....
905; .....
906; .....
907; .....
908; .....
909; .....
910; .....
911; .....
912; .....
913; .....
914; .....
915; .....
916; .....
917; .....
918; .....
919; .....
920; .....
921; .....
922; .....
923; .....
924; .....
925; .....
926; .....
927; .....
928; .....
929; .....
930; .....
931; .....
932; .....
933; .....
934; .....
935; .....
936; .....
937; .....
938; .....
939; .....
940; .....
941; .....
942; .....
943; .....
944; .....
945; .....
946; .....
947; .....
948; .....
949; .....
950; .....
951; .....
952; .....
953; .....
954; .....
955; .....
956; .....
957; .....
958; .....
959; .....
960; .....
961; .....
962; .....
963; .....
964; .....
965; .....
966; .....
967; .....
968; .....
969; .....
970; .....
971; .....
972; .....
973; .....
974; .....
975; .....
976; .....
977; .....
978; .....
979; .....
980; .....
981; .....
982; .....
983; .....
984; .....
985; .....
986; .....
987; .....
988; .....
989; .....
990; .....
991; .....
992; .....
993; .....
994; .....
995; .....
996; .....
997; .....
998; .....
999; .....
1000; .....
```

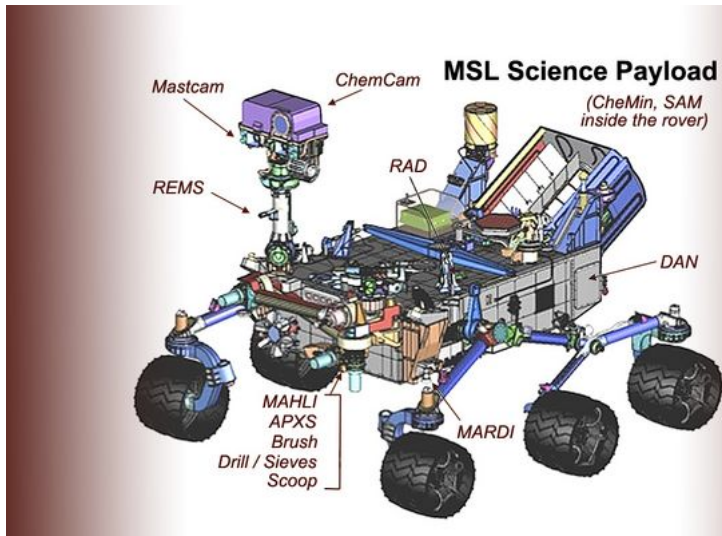


Aplicações

Curiosity Rover - NASA

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação





Aplicações

Curiosity Rover - NASA

Armazenamento
e organização
de dados em
Assembly

Daniel Moraes
- Engenharia
de
Computação

